



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΧΗΜΙΚΩΝ ΜΗΧΑΝΙΚΩΝ
ΤΟΜΕΑΣ ΑΝΑΛΥΣΗΣ, ΣΧΕΔΙΑΣΜΟΥ ΚΑΙ
ΑΝΑΠΤΥΞΗΣ ΔΙΕΡΓΑΣΙΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Επίλυση διαφορικών εξισώσεων
με μεθόδους μηχανικής μάθησης

ΦΩΤΟΠΟΥΛΟΣ ΓΕΩΡΓΙΟΣ

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΚΑΒΟΥΣΑΝΑΚΗΣ ΜΙΧΑΗΛ

ΑΘΗΝΑ, ΦΕΒΡΟΥΑΡΙΟΣ 2022

ΕΥΧΑΡΙΣΤΙΕΣ:

Θα ήθελα να ευχαριστήσω τον κύριο Καβουσανάκη Μιχαήλ, επιβλέποντα της παρούσης εργασίας, για τη δυνατότητα που μου έδωσε να πραγματοποιήσω τη συγκεκριμένη διπλωματική, το ενδιαφέρον που επέδειξε κατά τη διάρκεια της εκπόνησής της καθώς και τη βοήθεια και το χρόνο που μου προσέφερε.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΙΛΗΨΗ	i.
ABSTRACT	ii.
ΕΙΣΑΓΩΓΗ	iii.
ΚΕΦΑΛΑΙΟ ΠΡΩΤΟ	1.
1. Περί διαφορικών εξισώσεων	2.
1.1 Το πρόβλημα της διατύπωσης διαφορικών εξισώσεων	2.
1.2 Η λύση του προβλήματος	4.
2. Περί μηχανικής μάθησης.....	4.
2.1 Διεργασίες Gauss.....	5.
2.2 Τεχνητά νευρωνικά δίκτυα.....	6.
ΚΕΦΑΛΑΙΟ ΔΕΥΤΕΡΟ	11.
1. Συνοπτική περιγραφή της μεθοδολογίας.....	12.
2. Προσέγγιση συνάρτησης με επιλογή κατάλληλων παραμέτρων.....	12.
3.1. Πρόβλημα εύρεσης μεταβατικής κατάστασης συστήματος μερικών διαφορικών εξισώσεων	15.
3.2. Πρόβλημα εύρεσης μόνιμης κατάστασης συστήματος μερικών διαφορικών εξισώσεων	19.
4. Εφαρμογή μεθοδολογίας σε περιβάλλον Matlab	20.
ΚΕΦΑΛΑΙΟ ΤΡΙΤΟ	24.
1. Προσέγγιση συνάρτησης με επιλογή παραμέτρων και επίλυση διαφορικής εξίσωσης.....	25.
1.1 Αποτελέσματα εφαρμογής διεργασιών Gauss.....	25.
1.2 Αποτελέσματα προσέγγισης δεξιού σκέλους διαφορικής εξίσωσης και εφαρμογής ode solvers	25.
2. Υπολογισμός μεταβατικής κατάστασης για διαφορετικές αρχικές συνθήκες, σε πρόβλημα αυλωτού αντιδραστήρα και σύγκριση με δεδομένα από Comsol	30.
2.1 Σύγκριση αποτελεσμάτων μεταξύ μοντέλου NN και Comsol για τιμή παραμέτρου $Da = 0,12$	30.
2.2 Σύγκριση αποτελεσμάτων μεταξύ μοντέλου NN και Comsol για τιμή παραμέτρου $Da = 0,118$	32.
3. Υπολογισμός μόνιμης κατάστασης σε περιοχή πολλαπλότητας λύσεων, σε πρόβλημα αυλωτού αντιδραστήρα και σύγκριση με δεδομένα από Comsol	33.
ΚΕΦΑΛΑΙΟ ΤΕΤΑΡΤΟ	36.
ΒΙΒΛΙΟΓΡΑΦΙΑ	38.
ΠΑΡΑΡΤΗΜΑ	39.
Παράρτημα Α	39.
Παράρτημα Β	44.
Παράρτημα Γ	47.
Παράρτημα Δ	50.

ΠΕΡΙΛΗΨΗ

Στην παρούσα διπλωματική εργασία μελετήθηκε ο συνδυασμός μεθόδων μηχανικής μάθησης και συγκεκριμένα διεργασιών Gauss (Gaussian processes) και τεχνητών νευρωνικών δικτύων (artificial neural networks), με κλασσικές μεθόδους αριθμητικής ανάλυσης και επίλυσης διαφορικών εξισώσεων με στόχο την εκμάθηση της σχέσης που συνδέει γνωστά μεγέθη, που αποτελούν λύσεις διαφορικών εξισώσεων, με τις χρονικές παραγώγους των εξαρτημένων μεταβλητών. Έτσι, αν και είναι άγνωστη στο χρήστη η διατύπωση μίας διαφορικής εξίσωσης, αποδεικνύεται ότι εφαρμόζοντας τη συγκεκριμένη προσέγγιση (data-driven approach), διαδικασίες όπως η εύρεση μονίμων καταστάσεων και πολλαπλότητας λύσεων καθώς και η ολοκλήρωση σε μεγάλους χρόνους για εύρεση μεταβατικών καταστάσεων είναι εφικτές.

Συγκεκριμένα εξετάστηκαν δύο συστήματα. Στο πρώτο, μία απλή διαφορική εξίσωση, εφαρμόστηκαν διεργασίες Gauss προκειμένου να βρεθεί ποιες από τις υπό εξέταση μεταβλητές επηρεάζουν τη χρονική παράγωγο της εξαρτημένης μεταβλητής. Έπειτα, με εκπαίδευση μοντέλου τεχνητού νευρωνικού δικτύου για 7.000 σημεία προέκυψε μοντέλο, το οποίο δοκιμάστηκε με χρήση επιλύτη διαφορικών εξισώσεων (ode solver), επιβάλλοντας διαφορετικές αρχικές και συνοριακές συνθήκες από αυτές που χρησιμοποιήθηκαν για παραγωγή δεδομένων. Το εκπαιδευμένο μοντέλο βρέθηκε να παρουσιάζει αποκλίσεις μικρότερες από 5 % ως προς τις πραγματικές λύσεις ενώ, το μέσο τετραγωνικό σφάλμα ήταν σε κάθε περίπτωση της τάξης του 10^{-7} οδηγώντας στο συμπέρασμα ότι το εκπαιδευμένο μοντέλο δεν επηρεάζεται από το είδος των συνοριακών και αρχικών συνθηκών.

Στο δεύτερο παράδειγμα, μελετήθηκε ένα σύστημα δύο εξισώσεων με δύο εξαρτημένες μεταβλητές και μία παράμετρο, επαναλήφθηκε η διαδικασία εκπαίδευσης χρησιμοποιώντας 160.000 και 87.000 σημεία αντίστοιχα για την παραγωγή δύο μοντέλων. Το πρώτο χρησιμοποιήθηκε για την εύρεση της μεταβατικής κατάστασης του συστήματος μέσω ode solver, δοκιμαζόμενο σε επτά διαφορετικές αρχικές συνθήκες επιτυγχάνοντας ακρίβεια έως και πάνω από 98 % για χρόνο 40 φορές μεγαλύτερο από το μέγιστο για τον οποίο παρήχθησαν δεδομένα εκπαίδευσης. Το δεύτερο μοντέλο εκπαιδεύτηκε με τα δεδομένα εκείνα τα οποία αντιστοιχούσαν σε τιμές παραμέτρου τέτοιες, ώστε οι τιμές της μόνιμης κατάστασης του συστήματος να βρίσκονται σε ένα μόνο κλάδο ευσταθών λύσεων. Πραγματοποιήθηκαν έπειτα δοκιμές εφαρμόζοντας μέθοδο Newton-GMRES, για εύρεση της μόνιμης κατάστασης σε τιμή της παραμέτρου όπου το σύστημα αναμενόταν να εμφανίζει τρεις μόνιμες καταστάσεις, δύο ευσταθείς και μία ασταθή. Τα αποτελέσματα που προέκυψαν αποδεικνύουν ότι το μοντέλο, αν και με μεγάλη σχετικά απόκλιση για κάποιες τιμές (έως και 54 %), ήταν ικανό να προβλέψει την ύπαρξη τριών μόνιμων καταστάσεων.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Μερικές διαφορικές εξισώσεις, επιλύτες διαφορικών εξισώσεων, διεργασίες Gauss, τεχνητά νευρωνικά δίκτυα, μεταβατικό πρόβλημα, πρόβλημα μόνιμης κατάστασης, αριθμητική ανάλυση

ABSTRACT

In the present diploma thesis, the combination of machine learning techniques, namely Gaussian processes and artificial neural networks, with classic numerical analysis and differential equation solving methods, has been studied, with the goal of mapping the relationship between known quantities and the time derivatives of dependent variables of differential equations. This way, despite the fact that the formulation of a differential equation is unknown, following the data-driven approach presented in this work, one can calculate solutions for both transient and steady-state problems, including ones with solution multiplicity.

Two examples are presented. The first is a relatively simple, partial differential equation. Gaussian processes were applied in order to discover the variables that affect the time derivative. Using 7.000 points, a neural network was trained and tested using ordinary differential equation solvers, for different initial and boundary conditions than the ones used to produce training data. The trained model was found to produce solutions, deviating from the true ones by less than 5 %. The mean squared error in every simulation was of the 10^{-7} order, proving that the trained model is not affected by the kind of boundary and initial conditions used.

In the second example, a system of two partial differential equations describing the operation of a tubular reactor was studied. The system comprised of two independent variables and one parameter. Two neural network models were trained using 160.000 and 87.000 points respectively. The first model was utilized for the calculation of transient solutions, for seven different initial conditions, achieving up to 98 % accuracy in times 40 times greater than the maximum used to produce training data. The second model was trained using data corresponding to a single stable solutions' branch. It was then tested, using Newton-GMRES numerical method, for parameter value such, as to ensure the existence of three different steady states. It was proven that the trained model was in position of calculating all possible solutions, despite errors of up to 54 %.

KEYWORDS: Partial differential equations, ode solvers, Gaussian processes, artificial neural networks, transient problem, steady-state problem, numerical analysis

ΕΙΣΑΓΩΓΗ

Στόχος της παρούσης εργασίας

Αντικείμενο της παρούσης διπλωματικής εργασίας αποτελεί η εφαρμογή μεθόδων μηχανικής μάθησης και συγκεκριμένα διεργασιών Gauss και τεχνητών νευρωνικών δικτύων, για την επιλογή των κατάλληλων παραμέτρων που επηρεάζουν μία διεργασία και την ανάπτυξη μοντέλων που θα προσεγγίζουν το δεξί σκέλος διαφορικής εξίσωσης (ή συστήματος εξισώσεων). Αν και μεθοδολογίες οι οποίες προβλέπουν το δεξί σκέλος διαφορικής εξίσωσης έχουν αναπτυχθεί από ερευνητές στο παρελθόν (π.χ. Lee et al., 2020, Raisi & Karniadakis, 2018), εν προκειμένω τα εκπαιδευμένα μοντέλα χρησιμοποιούνται για την παραγωγή λύσεων εφαρμόζοντας κλασσικές μεθόδους αριθμητικής ανάλυσης, όπως η μέθοδος πεπερασμένων διαφορών, επιλύτες συνήθων διαφορικών εξισώσεων (ode solvers) και η μέθοδος Newton-GMRES. Επιχειρείται ναδειχθεί ότι, εφόσον ένα εκπαιδευμένο μοντέλο έχει ‘γνώση’ της σχέσης που συνδέει το δεξί σκέλος μίας διαφορικής εξίσωσης, δηλαδή τις ανεξάρτητες μεταβλητές, με το αριστερό σκέλος, δηλαδή το χρονικό διαφορικό της εξαρτημένης μεταβλητής, μπορεί να προβλέπει με ακρίβεια μεταβατική κατάσταση σε χρόνους όπου δεν υπάρχουν δεδομένα. Επίσης είναι ικανό να προβλέπει μόνιμη κατάσταση και πολλαπλότητα λύσεων. Κατ’ ουσίαν, τα εκπαιδευμένα μοντέλα μπορούν να χρησιμοποιηθούν για επίλυση μίας διαφορικής εξίσωσης η οποία δεν είναι γνωστή. Έτσι μπορούν να χρησιμοποιηθούν σε περιπτώσεις όπου υπάρχει άγνοια περί των διαφορικών εξισώσεων που διέπουν ένα σύστημα και των παραμέτρων που το επηρεάζουν. Επίσης η μεθοδολογία της παρούσης εργασίας μπορεί να εφαρμοστεί σε περιπτώσεις όπου υπάρχουν πειραματικά δεδομένα, χωρίς να μπορεί να διατυπωθεί με ακρίβεια μοντέλο που να τα περιγράφει.

Διάρθρωση διπλωματικής

Στο πρώτο κεφάλαιο της παρούσης διπλωματικής αναφέρονται βασικά πράγματα περί διαφορικών εξισώσεων, αρχικών και συνοριακών συνθηκών. Αναλύεται το κίνητρο για την ανάπτυξη της μεθοδολογίας της διπλωματικής ενώ περιγράφονται αρχές της μηχανικής μάθησης καθώς και στοιχειώδης θεωρία για τις διεργασίες Gauss και τα τεχνητά νευρωνικά δίκτυα.

Στο δεύτερο κεφάλαιο αναπτύσσεται αναλυτικά η μεθοδολογία που ακολουθείται για την παραγωγή αποτελεσμάτων της διπλωματικής. Χωρίζεται σε τέσσερις ενότητες. Η πρώτη περιλαμβάνει την δημιουργία δεδομένων για την εκπαίδευση μοντέλου νευρωνικού δικτύου το οποίο εφαρμόζεται σε συνδυασμό με επιλύτη διαφορικών εξισώσεων, προκειμένου να γίνει πρόβλεψη των λύσεων διαφορικής εξίσωσης. Εξετάζεται το κατά πόσο το είδος των αρχικών και συνοριακών συνθηκών επιδρούν στο μοντέλο. Στη δεύτερη ενότητα, ακολουθείται όμοια διαδικασία αλλά για σύστημα πλέον δύο εξαρτημένων μεταβλητών με πιο περίπλοκη δυναμική. Στην τρίτη ενότητα περιγράφεται η διαδικασία υπολογισμού της μόνιμης κατάστασης χρησιμοποιώντας τα μοντέλα που κατασκευάστηκαν προηγουμένως. Στην τελευταία ενότητα του κεφαλαίου αυτού γίνεται αναφορά στον τρόπο εφαρμογής της μεθοδολογίας μέσω Matlab.

Στο τρίτο κεφάλαιο παρουσιάζονται τα αποτελέσματα σε μορφή διαγραμμάτων καθώς και τα σχετικά σφάλματα σε μορφή πινάκων. Αποδεικνύεται η ικανότητα των εκπαιδευμένων μοντέλων για ακριβή πρόβλεψη σε χρόνους μέχρι και σαράντα φορές μεγαλύτερους από το μέγιστο χρόνο για τον οποίο υπάρχουν διαθέσιμα δεδομένα. Επίσης αποδεικνύεται η δυνατότητα του μοντέλου να προβλέπει πολλαπλότητα λύσεων χωρίς να έχει εκπαιδευτεί με δεδομένα σε τιμές παραμέτρου όπου παρουσιάζεται πολλαπλότητα.

Στο σύντομο τέταρτο κεφάλαιο αναφέρονται τα συμπεράσματα που προκύπτουν από την εν λόγω διπλωματική καθώς και προτάσεις για μελλοντικές εργασίες.

Τέλος παρατίθενται η βιβλιογραφία καθώς και παράρτημα στο οποίο περιγράφεται η διαδικασία δημιουργίας και εξαγωγής δεδομένων σε περιβάλλον COMSOL Multiphysics και οι κώδικες που αναπτύχθηκαν για την διενέργεια των δοκιμών της διπλωματικής.

ΚΕΦΑΛΑΙΟ ΠΡΩΤΟ

Θεωρητικό μέρος

Περί διαφορικών εξισώσεων, μηχανικής μάθησης, διεργασιών Gauss και τεχνητών
νευρωνικών δικτύων

1. Περί διαφορικών εξισώσεων

Διαφορικές ονομάζονται οι εξισώσεις εκείνες, οι οποίες εκφράζουν σχέσεις μεταξύ συναρτήσεων και των παραγώγων τους. Απαντώνται σε πλήθος εφαρμογών της Μηχανικής όπως για παράδειγμα κατά τη διατύπωση ισοζυγίων μάζας και ενέργειας. Οι διαφορικές εξισώσεις μπορούν να ταξινομηθούν με διάφορα κριτήρια όπως για παράδειγμα την τάξη των παραγώγων (η τάξη της ανώτερης παραγώγου αποτελεί τάξη της εξίσωσης) ή την ύπαρξη μη γραμμικών όρων (εξισώσεις που περιέχουν μόνο γραμμικούς όρους ονομάζονται γραμμικές). Ένας βασικός τρόπος ταξινόμησης των διαφορικών εξισώσεων έγκειται στον αριθμό των ανεξάρτητων μεταβλητών από τις οποίες εξαρτάται η άγνωστη συνάρτηση. Στην περίπτωση μίας ανεξάρτητης μεταβλητής (η οποία συνήθως είναι ο χρόνος), οι εξισώσεις ονομάζονται συνήθεις διαφορικές (ordinary differential equations, ode's) και οι παράγωγοι αναγράφονται ως ολικά διαφορικά ενώ αν οι ανεξάρτητες μεταβλητές είναι περισσότερες (επί παραδείγματι ο χρόνος και μία ή περισσότερες χωρικές μεταβλητές) οι διαφορικές εξισώσεις χαρακτηρίζονται ως μερικές (partial differential equations, pde's) και οι παράγωγοι των συναρτήσεων αναγράφονται ως μερικά διαφορικά.

Χαρακτηριστικό παράδειγμα συνήθους διαφορικής εξίσωσης αποτελεί η παρακάτω εξίσωση:

$$\frac{dC_A}{dt} = -k \cdot C_A \quad (1.1)$$

η οποία εκφράζει το ρυθμό μεταβολής της συγκέντρωσης αντιδρώντος A, λόγω χημικής αντίδρασης με κινητική πρώτης τάξης, σταθεράς k σε αντιδραστήρα διαλείποντος έργου. Με ολοκλήρωση προκύπτει η παρακάτω εξίσωση:

$$C_A = C_{A,t=0} e^{-k \cdot t} \quad (1.2)$$

Προκειμένου να επιλυθεί η εξίσωση (1.1), να βρεθεί δηλαδή η συγκέντρωση αντιδρώντος A, C_A , σε μία χρονική στιγμή t, απαιτείται η γνώση της τιμής της μεταβλητής τη χρονική στιγμή $t=0$, $C_{A,t=0}$, δηλαδή μία αρχική συνθήκη (initial condition, IC). Ένα ακόμη παράδειγμα συνήθους διαφορικής εξίσωσης αποτελεί η εξίσωση Laplace, η οποία χαρακτηρίζει μονοδιάστατη αγωγή θερμότητας σε μόνιμη κατάσταση, μέσω επίπεδου τοιχώματος μεγάλου μήκους χωρίς παραγωγή θερμότητας:

$$\frac{d^2T}{dx^2} = 0, x \in (0,1) \quad (1.3)$$

Η λύση της παραπάνω εξίσωσης συνδέει τη θερμοκρασία (T) με την απόσταση από το άκρο του τοιχώματος (x) και είναι:

$$T = c_1 x + c_2 \quad (1.4)$$

Συνεπώς, για να επιλυθεί η (1.3), δηλαδή να υπολογιστούν οι σταθερές c_1 και c_2 της (1.4), χρειάζονται δύο τιμές της θερμοκρασίας σε δύο σημεία του τοιχώματος. Τα σημεία αυτά είναι συνήθως τα ακραία (εν προκειμένω $x = 0$ και $x = 1$) και οι βοηθητικές αυτές τιμές ονομάζονται συνοριακές συνθήκες (boundary conditions, BC). Υπάρχουν τρία είδη συνοριακών συνθηκών. Η συνθήκη Dirichlet (ή πρώτου είδους) έχει τη μορφή $T(x_0) = c$ και δηλώνει ότι η τιμή της μεταβλητής y στο σημείο x_0 παραμένει σταθερή. Ως παράδειγμα αναφέρεται η συνθήκη μη ολίσθησης σύμφωνα με την οποία η ταχύτητα ενός ρευστού σε επαφή με στερεό είναι ίση με την ταχύτητα του στερεού. Η συνθήκη Neumann (ή δευτέρου είδους) έχει την ακόλουθη μορφή:

$$\left. \frac{dT}{dx} \right|_{x_0} = c \quad (1.5)$$

και δηλώνει ότι ο ρυθμός μεταβολής στο σημείο x_0 διατηρείται σταθερός. Παράδειγμα αποτελεί η μηδενική ροή θερμότητας μέσω πλήρως μονωμένου τοιχώματος. Η συνθήκη Robin (ή τρίτου είδους) έχει τη μορφή συνάρτησης:

$$f\left(\left. \frac{dT}{dx} \right|_{x_0}, T(x_0)\right) = 0 \quad (1.6)$$

και χρησιμοποιείται για την περιγραφή ρυθμού μεταβολής εξαρτημένου από την τιμή της μεταβλητής y στο σημείο x_0 . Ένα τέτοιο παράδειγμα αποτελεί η παρακάτω εξίσωση η οποία εμφανίζεται στη μελέτη φαινομένων μεταφοράς και εκφράζει το ρυθμό μεταφοράς θερμότητας με αγωγή σε ελεύθερο άκρο τοιχώματος, ο οποίος ισούται με τη ροή θερμότητας με συναγωγή από το περιβάλλον:

$$k \left. \frac{dT}{dx} \right|_{x_0} = h(T(x_0) - T_e) \quad (1.7)$$

Όσον αφορά την επίλυση διαφορικών εξισώσεων με περισσότερες της μίας ανεξαρτήτων μεταβλητών, έστω η ακόλουθη μερική διαφορική εξίσωση:

$$\frac{\partial T}{\partial t} = \frac{\partial^2 T}{\partial x^2}, x \in (0,1) \quad (1.8)$$

Η παραπάνω εξίσωση περιγράφει το ίδιο πρόβλημα με την εξίσωση (1.3), στη μεταβατική όμως κατάσταση (ανεξάρτητες μεταβλητές ο χρόνος και μία χωρική μεταβλητή) και για να επιλυθεί χρειάζεται τόσο η γνώση μίας αρχικής συνθήκης, δηλαδή μίας τιμής του T για κάθε x τη στιγμή $t = t_0$, όσο και δύο συνοριακές συνθήκες, δηλαδή δύο τιμές του T σε δύο σημεία x_a, x_b σε κάθε χρονική στιγμή t . Η αρχική συνθήκη μπορεί να είναι ένα προφίλ (εν προκειμένω θερμοκρασίας) ενώ οι συνοριακές συνθήκες θα μεταβάλλονται με το χρόνο αν είναι Neumann ή Robin.

Η εύρεση των λύσεων μπορεί να γίνει αναλυτικά στην περίπτωση γραμμικών εξισώσεων (διαφορικές εξισώσεις που περιλαμβάνουν μόνο γραμμικούς όρους) αλλά σε κάθε περίπτωση μπορούν να χρησιμοποιηθούν αριθμητικές μέθοδοι με τη βοήθεια υπολογιστή. Για παράδειγμα μπορεί να εφαρμοστεί η μέθοδος Newton-Raphson πολλών μεταβλητών για την εύρεση των τιμών των ανεξαρτήτων μεταβλητών στη μόνιμη κατάσταση. Επιπλέον, με μεθόδους όπως η μέθοδος των γραμμών μπορεί να λυθεί το μεταβατικό πρόβλημα, να υπολογιστεί δηλαδή η τιμή της συνάρτησης σε κάθε χρονική στιγμή ή, εκφράζοντας διαφορετικά το πρόβλημα, να υπολογιστεί σε κάθε χρονική στιγμή το διάνυσμα $\partial T / \partial t$ (πρόκειται για διάνυσμα καθώς η παράγωγος $\partial T / \partial t$ θα παίρνει διαφορετική τιμή σε κάθε x). Η μερική παράγωγος ως προς το χρόνο $\partial v / \partial t$, με εξαρτημένη μεταβλητή την v , στο εξής θα αναφέρεται απλώς ως αριστερό σκέλος της διαφορικής εξίσωσης (left-hand side) ενώ θα γίνεται αναφορά σε δεξί σκέλος της διαφορικής εξίσωσης για τους όρους από τους οποίους εξαρτάται το αριστερό σκέλος (όπως $\partial^2 v / \partial x^2$, $\partial v / \partial x$, $g(v)$ κ.α.).

Όπως συμβαίνει και στην περίπτωση που μία διεργασία περιγράφεται από σύστημα αλγεβρικών εξισώσεων, οι διαφορικές εξισώσεις μπορεί επίσης να συναντώνται και σε συστήματα (συνήθων ή μερικών) διαφορικών εξισώσεων. Χαρακτηριστικό παράδειγμα αποτελούν οι εξισώσεις που περιγράφουν το πρόβλημα κυνηγού-θηράματος (Boyce, DiPrima, 1997):

$$\frac{dx}{dt} = x(\alpha - \beta \cdot y) \quad (1.9, \alpha)$$

$$\frac{dy}{dt} = y(-\gamma + \delta \cdot x) \quad (1.9, \beta)$$

Στο παραπάνω σύστημα εξισώσεων ανεξάρτητη μεταβλητή αποτελεί ο χρόνος t , με $\alpha, \beta, \gamma, \delta$ θετικές σταθερές.

1.1. Το πρόβλημα της διατύπωσης διαφορικών εξισώσεων

Οι διαφορικές εξισώσεις, συνήθεις ή μερικές απαντώνται όπως προαναφέρθηκε σε πλήθος προβλημάτων. Όμως υπάρχουν περιπτώσεις διεργασιών όπου, λόγω της περιπλοκότητας των φαινομένων που εμπλέκονται σε αυτές, δεν είναι δυνατό να διατυπωθούν μερικές διαφορικές εξισώσεις που να περιγράφουν μακροσκοπικά την εξέλιξή τους.

Μία τέτοια κατηγορία κλασικών προβλημάτων χημικής μηχανικής είναι αυτά που απαντώνται σε χημικές αντιδράσεις και αντιδραστήρες. Ως παράδειγμα αναφέρεται αυτό της χημικής απόθεσης από ατμό (CVD – Chemical Vapor Deposition). Στην περίπτωση της CVD, γίνεται εισαγωγή αερίων σε ένα θάλαμο απόθεσης (αντιδραστήρας) όπου βρίσκονται τοποθετημένες στερεές επιφάνειες επί των οποίων γίνεται απόθεση ενός λεπτού υμενίου (φιλμ) (Dobkin & Zuraw, 2003). Οι αντιδράσεις που λαμβάνουν χώρα στον αντιδραστήρα είναι ομογενείς (στην αέρια φάση) καθώς και ετερογενείς (στο στερεό όπου αναπτύσσεται το υμένιο). Η απόκτηση ακριβούς γνώσης για το ρυθμό απόθεσης είναι δύσκολη καθώς

τα φαινόμενα μεταφοράς μάζας και θερμότητας (π.χ. διάχυση αντιδρώντων στο αέριο, διάχυση αντιδρώντων και προϊόντων στο στερεό υμένιο, ροή αερίου, μεταφορά θερμότητας με αγωγή εντός του στερεού και συναγωγή στο αέριο, μεταβολές θερμοκρασίας λόγω εξώθερμων αντιδράσεων) όπως και τα φυσικοχημικά φαινόμενα σε υψηλές θερμοκρασίες (π.χ. κινητικές αντιδράσεων, δημιουργία πλάσματος) είναι ιδιαίτερα πολύπλοκα. Αυτό μεταφράζεται σε αδυναμία διατύπωσης διαφορικής εξίσωσης, αριστερό σκέλος της οποίας θα αποτελούσε ο ρυθμός αύξησης του πάχους υμενίου και δεξί οποιαδήποτε συνάρτηση συγκεντρώσεων και θερμοκρασιών αντιδρώντων, ρυθμών ροής και άλλων μακροσκοπικών παραμέτρων.

Η διεργασία μπορεί να βελτιστοποιηθεί βάσει πειραμάτων και εμπειρίας, κάτι που αποτελεί σε ορισμένες περιπτώσεις την πιο εύκολη επιλογή. Όμως η περιπλοκότητα ορισμένων διεργασιών CVD καθιστά μη αποδοτική την εμπειρική προσέγγιση (Pierson, 1999).

Γενικά, η παραγωγή διαφορετικών εξισώσεων που διέπουν μακροσκοπικά τη λειτουργία ενός συστήματος αποτελεί μία χρονοβόρο διαδικασία που απαιτεί βαθιά γνώση του συστήματος (Lee et al., 2020). Επίσης, σημαντική δυσκολία μπορεί να αποτελεί η επιλογή των κατάλληλων μεταβλητών (Lee et al., 2020) που θα υπάρχουν στο δεξί σκέλος των μακροσκοπικών διαφορικών εξισώσεων.

1.2. Η λύση του προβλήματος

Πράγματι, στο παράδειγμα της CVD, θα ήταν χρήσιμο να υπήρχε τρόπος επιλογής των παραμέτρων που επηρεάζουν περισσότερο τη διεργασία, οι οποίες θα χρησιμοποιούνταν για την κατασκευή μοντέλου που θα υπολόγιζε το ρυθμό αύξησης του πάχους ενός φιλμ ημιαγωγού, λαμβάνοντας υπόψη μόνο αυτές τις παραμέτρους, χωρίς την ανάγκη για ακριβή γνώση των συνθηκών που επικρατούν σε κάθε σημείο του αντιδραστήρα. Μία λύση σε τέτοιου είδους προβλήματα δίνεται με εφαρμογή μεθόδων μηχανικής μάθησης, με τις οποίες πραγματοποιείται στα πλαίσια της παρούσης εργασίας, τόσο η επιλογή παραμέτρων όσο και η κατασκευή μοντέλου που θα προβλέπει το αριστερό σκέλος μίας άγνωστης διαφορικής εξίσωσης (ή ενός συστήματος αγνώστων διαφορικών εξισώσεων). Όπως αναφέρεται και παρακάτω εικάζεται ότι θα υπάρχει κάποια διαφορική εξίσωση που να περιγράφει το σύστημα και η οποία θα προσεγγίζεται από μοντέλο μηχανικής μάθησης. Πρόκειται για μία προσέγγιση η οποία βασίζεται σε δεδομένα (data-driven approach) (Arbabi & Kevrekidis, 2021, Raissi & Karniadakis, 2018).

2. Περί μηχανικής μάθησης

Κατά τη διάρκεια της ζωής του, ένας άνθρωπος, ζώντας σε ένα διαρκώς μεταβαλλόμενο περιβάλλον, βρίσκεται αντιμέτωπος με πολλές διαφορετικές καταστάσεις από τις οποίες αποκτά εμπειρία. Αξιοποιώντας την εμπειρία μπορεί να προσαρμόζεται και να βελτιώνεται. Κατά αναλογία, με τον όρο μηχανική μάθηση νοείται η ικανότητα ενός υπολογιστικού συστήματος να βελτιώνεται αξιοποιώντας την εμπειρία. Πιο αυστηρά, ως πρόβλημα μάθησης ορίζεται, “το πρόβλημα της βελτίωσης του μέτρου της απόδοσης ενός συστήματος στην εκτέλεση μίας διεργασίας, μέσω κάποιου είδους εμπειρίας” (Jordan & Mitchell, 2015). Το σύστημα, στην περίπτωση της μηχανικής μάθησης, είναι υπολογιστικό ενώ ο όρος “εμπειρία” στον οποίο έγινε αναφορά, δεν αποτελεί παρά δεδομένα. Τα δεδομένα αυτά μπορεί να είναι διαφόρων ειδών και ποικίλουν ανάλογα με την εφαρμογή.

Με βάση τον ορισμό που δόθηκε παραπάνω, ως μάθηση μπορεί να χαρακτηριστεί η ανάπτυξη απλών μοντέλων γραμμικής παλινδρόμησης (συστήματα), χρησιμοποιώντας ζεύγη τιμών εξαρτημένης και ανεξάρτητης μεταβλητής (εμπειρία), για την πρόβλεψη (διεργασία) αποτελεσμάτων σε περιοχές όπου δεν υπάρχουν δεδομένα. Η ανάλυση δεδομένων καθώς και η προσαρμογή σε αυτά ενός μοντέλου, αποτελούσε έργο των επιστημόνων πολύ πριν την εμφάνιση των ηλεκτρονικών υπολογιστών.

Όμως, δεν μπορούν όλα τα μοντέλα να είναι απλές γραμμικές ή εκθετικές συναρτήσεις. Ένα τέτοιο παράδειγμα, αυτό της χημικής απόθεσης από ατμό, δόθηκε παραπάνω. Όμοια, έστω το παράδειγμα της εκτίμησης της τιμής πώλησης ενός ακινήτου σε μία πόλη, από ανάλυση δεδομένων προηγούμενων πωλήσεων. Ενδέχεται να υπάρχουν πολλοί παράγοντες οι οποίοι καθορίζουν την τιμή πώλησης. Η παλαιότητα του ακινήτου καθώς και η έκτασή του αναμένεται να είναι κάποιοι από αυτούς. Ωστόσο, μπορεί να επηρεάζεται από πολλούς άλλους παράγοντες, όπως η περιοχή στην οποία βρίσκεται, η πρόσβαση σε συγκοινωνίες και δημόσιες υπηρεσίες και άλλοι. Για να δημιουργηθούν κανόνες για μια τέτοια εκτίμηση απαιτείται χρονοβόρα εργασία και η εφαρμογή τους δεν είναι σίγουρο ότι θα έχει την επιθυμητή ακρίβεια.

Ένα ακόμη θέμα στο οποίο γίνεται αναφορά, είναι ο μεγάλος όγκος δεδομένων προς επεξεργασία. Η μεγάλη διαθεσιμότητα φθηνών μέσων δημιουργίας δεδομένων (όπως αισθητήρες) και αποθήκευσης

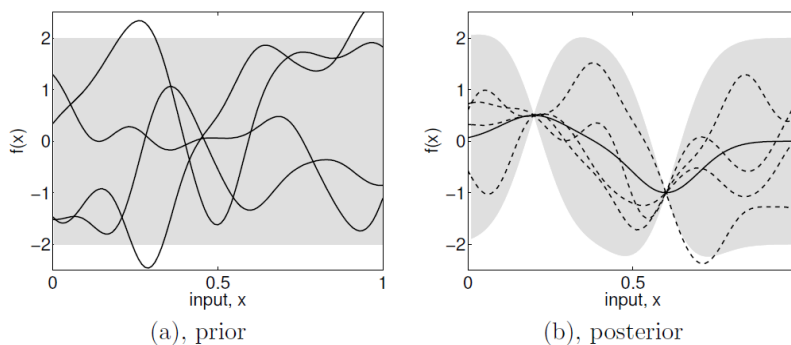
(όπως σκληρών δίσκων) καθώς και η δυνατότητα αποθήκευσης δεδομένων στο διαδίκτυο έχουν αυξήσει και συνεχίζουν να αυξάνουν τον όγκο των διαθέσιμων δεδομένων. Μάλιστα, υπάρχει χάσμα μεταξύ ρυθμού δημιουργίας δεδομένων και εξαγωγής χρήσιμων μοτίβων που προκύπτουν από αυτά (Witten et al, 2017). Οι παραπάνω παράγοντες, σε συνδυασμό με την ανάπτυξη και ραγδαία πρόοδο των ηλεκτρονικών υπολογιστών, με διαρκώς αυξανόμενες δυνατότητες επεξεργασίας δεδομένων και χαμηλό σχετικά κόστος, είναι αυτοί που έχουν δώσει τα τελευταία χρόνια σημαντική ώθηση και έχουν καταστήσει δημοφιλή τον κλάδο της μηχανικής μάθησης, παρά το γεγονός ότι πολλά από τα εργαλεία που αξιοποιεί αναπτύχθηκαν πριν αρκετές δεκαετίες (για τις διεργασίες Gauss που χρησιμοποιούνται στα πλαίσια της παρούσης εργασίας αναφέρεται (Rasmussen & Williams, 2005) ότι χρησιμοποιούνται ήδη από τα τέλη του 19^{ου} αιώνα).

Από τα προηγούμενα, γίνεται αντιληπτό πως οι μέθοδοι μηχανικής μάθησης εξυπηρετούν την εξαγωγή συμπερασμάτων, όπως για παράδειγμα των μεταβλητών που είναι χρήσιμες στη μελέτη μίας διεργασίας, από ένα μεγάλο πλήθος διαθέσιμων δεδομένων που δεν θα ήταν εφικτό ή οικονομικό να αναλυθούν από ανθρώπους. Οι μέθοδοι μηχανικής μάθησης βασίζονται στην αντίληψη πως, αν και δεν είναι γνωστός ο τρόπος που συσχετίζονται ορισμένα δεδομένα (στα πλαίσια της συγκεκριμένης εργασίας η συσχέτιση δεν είναι παρά η διαφορική εξίσωση), η σχέση αυτή δεν μπορεί να είναι τυχαία. Για παράδειγμα, σε μία εφαρμογή αναγνώρισης χειρόγραφου κειμένου από σύστημα μηχανικής μάθησης, κάθε χαρακτήρας δεν αποτελεί τυχαίο άθροισμα σημείων αλλά ακολουθεί κάποια μοτίβα.

Οι μέθοδοι μηχανικής μάθησης μπορούν να κατηγοριοποιηθούν βάσει διαφόρων κριτηρίων. Ένα βασικό κριτήριο αποτελεί το είδος του προβλήματος, για την επίλυση του οποίου επιστρατεύονται. Παραδείγματα αποτελούν τα προβλήματα κατηγοριοποίησης ή ταξινόμησης (classification) και τα προβλήματα παλινδρόμησης (regression). Οι δύο παραπάνω κατηγορίες προβλημάτων απαιτούν για την επίλυσή τους εκπαιδευτή ή 'επόπτη' ο οποίος θα παρέχει την τιμή της μεταβλητής εξόδου για κάθε διάνυσμα τιμών μεταβλητών εισόδου, ώστε να χαρτογραφηθεί η μεταξύ τους σχέση (supervised learning). Αυτή είναι η πλέον κοινή μορφή μηχανικής μάθησης που απαντάται στις διάφορες εφαρμογές (LeCun, Bengio & Hinton 2015). Σε αντίθετη περίπτωση, μόνο μεταβλητές εισόδου είναι διαθέσιμες (unsupervised learning) και στόχο αποτελεί η εύρεση μοτίβων. Οι μέθοδοι μηχανικής μάθησης που αξιοποιούνται στην παρούσα διπλωματική (δηλαδή διεργασίες Gauss και τεχνητά νευρωνικά δίκτυα), χρησιμοποιούνται ως μέθοδοι παλινδρόμησης με σκοπό την εύρεση μίας σχέσης μεταξύ των μεταβλητών εισόδου και εξόδου.

2.1. Διεργασίες Gauss (Gaussian Processes - GP)

Οι διεργασίες Gauss χρησιμοποιούνται στα πλαίσια της παρούσης διπλωματικής εργασίας, ως εργαλεία παλινδρόμησης (Gaussian Process Regression - GPR). Αποτελούν γενίκευση της κανονικής κατανομής Gauss σε χώρο απείρων διαστάσεων και εκφράζουν την πυκνότητα πιθανότητας για κάθε πιθανή συνάρτηση που διέρχεται από σημεία, που χρησιμοποιούνται για εκπαίδευση (Rasmussen & Williams, 2006). Μία γεωμετρική τους ερμηνεία φαίνεται στο ακόλουθο διάγραμμα:



Διάγραμμα 1.1. Γεωμετρική ερμηνεία διεργασίας Gauss. (a) πριν και (b) μετά από εκπαίδευση. Πηγή: (Rasmussen & Williams, 2006)

Στο παραπάνω διάγραμμα, η διεργασία Gauss ξεκινά ως μία πυκνότητα από άπειρα σημεία πιθανών συναρτήσεων, οι οποίες περιορίζονται (b) με την παροχή σημείων από τα οποία οφείλουν να διέρχονται οι συναρτήσεις αυτές.

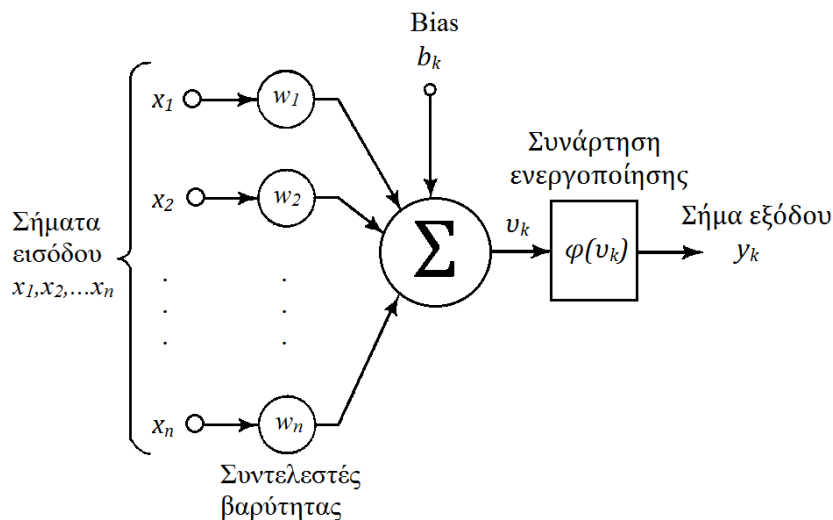
Όπως και οι κανονικές κατανομές σε μία διάσταση, οι διεργασίες Gauss, για να ορίζονται πλήρως απαιτούν δύο στοιχεία, μία συνάρτηση μέσου, η οποία συνήθως λαμβάνεται ίση με το μηδέν καθώς και μία συνάρτηση απόκλισης (covariance matrix ή kernel, K). Στην περίπτωση που η διεργασία Gauss πρόκειται να χρησιμοποιηθεί ως εργαλείο για την ανακάλυψη των μεταβλητών που επηρεάζουν εντονότερα ένα αποτέλεσμα, ως συνάρτηση kernel χρησιμοποιείται η radial basis function (RDF ή ARD) (Lee et al., 2020, Raissi & Karniadakis, 2018):

$$K_{ij} = \theta_o \cdot \exp\left(-\frac{1}{2} \sum_{l=1}^n \frac{(x_{i,l} - x_{j,l})^2}{\theta_l}\right) \quad (1.10)$$

Στην παραπάνω εξίσωση είναι $\theta = [\theta_o, \theta_1, \dots, \theta_k]^T$ ένα διάνυσμα μήκους $(k+1)$ με k τις διαστάσεις, ήτοι το πλήθος, των δεδομένων εισόδου. Οι διεργασίες Gauss αποτελούν μη παραμετρικά εργαλεία μηχανικής μάθησης, σε αντίθεση με τα τεχνητά νευρωνικά δίκτυα τα οποία περιγράφονται παρακάτω. Η διαδικασία μάθησης έγκειται στον υπολογισμό τιμών του διανύσματος θ έτσι, ώστε να ελαχιστοποιείται ο αρνητικός λογάριθμος της οριακής πιθανότητας (log marginal likelihood). Οι απαιτούμενοι υπολογισμοί είναι πολύπλοκοι και περιλαμβάνουν την αντιστροφή ενός πίνακα $n \times n$, διεργασία με υπολογιστικό κόστος $O(n^3)$. Για το λόγο αυτό ο αριθμός των σημείων που θα χρησιμοποιηθούν για εκπαίδευση περιορίζεται σημαντικά.

2.2. Τεχνητά νευρωνικά δίκτυα (Artificial Neural Networks – ANN ή απλώς NN)

Τα τεχνητά νευρωνικά δίκτυα (στο εξής θα αναφέρονται απλώς ως νευρωνικά δίκτυα - NN) αποτελούν εργαλεία μηχανικής μάθησης και οφείλουν την ονομασία τους στην προσπάθεια ερευνητών να εξηγήσουν και να μιμηθούν τον τρόπο λειτουργίας και αλληλεπίδρασης μεταξύ βιολογικών νευρώνων. Βασική δομική μονάδα κάθε νευρωνικού δικτύου αποτελούν οι νευρώνες. Ένα μοντέλο νευρώνα φαίνεται στο σχήμα που ακολουθεί.



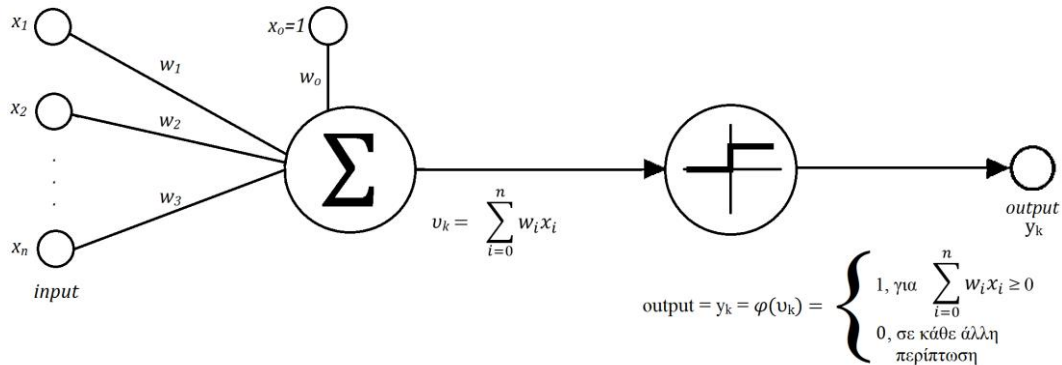
Σχήμα 1.1 . Απλό μοντέλο νευρώνα (Haykin, 2006).

Στο σχήμα 1.1, διακρίνονται τα τρία βασικά στοιχεία κάθε νευρώνα. Πρώτο βασικό στοιχείο αποτελούν οι συνάψεις, σε κάθε μία από τις οποίες ανατίθεται ένας συντελεστής βαρύτητας w_i (αναφέρονται στη διεθνή βιβλιογραφία ως synaptic weights ή απλώς weights). Τα γινόμενα των συντελεστών βαρύτητας με τα αντίστοιχα σήματα εισόδου x_i αθροίζονται εν συνεχεία σε ένα κόμβο (Σ) ο οποίος αποτελεί το δεύτερο βασικό στοιχείο κάθε νευρώνα. Έτσι προκύπτει ένα άθροισμα u_k το οποίο αποτελεί όρισμα της συνάρτησης ενεργοποίησης φ (τρίτο στοιχείο, αναφέρεται επίσης ως συνάρτηση μεταφοράς). Στόχος του συγκεκριμένου στοιχείου είναι ο περιορισμός του σήματος εξόδου σε συγκεκριμένο εύρος τιμών. Αναλυτικά:

$$y_k = \varphi(u_k), \text{ όπου } u_k = \sum_{i=1}^n w_i x_i + b_k \quad (1.11)$$

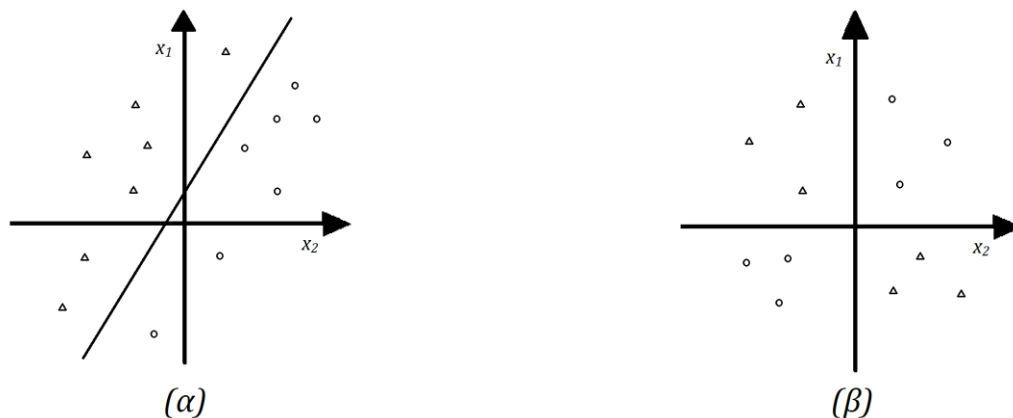
Στο σχήμα 1.1 εντοπίζεται επίσης ένας συντελεστής b_k ο οποίος αναφέρεται στην ξενόγλωσση βιβλιογραφία ως 'bias' και το σήμα του είναι ισοδύναμο με ένα συναπτικό βάρος w_0 για σταθερό σήμα εισόδου $x_0 = 1$. Σκοπός του συντελεστή αυτού είναι η οριζόντια μετατόπιση συνάρτησης ενεργοποίησης.

Ιστορικά, σταθμό στην εξέλιξη των νευρωνικών δικτύων αποτέλεσε η έννοια του perceptron (Rosenblatt, 1958). Γραφική απεικόνιση του perceptron παρουσιάζεται στο ακόλουθο σχήμα.



Σχήμα 1.2. Μοντέλο νευρώνα perceptron (Mitchell, 1997).

Χαρακτηριστικό του perceptron αποτελεί η συνάρτηση ενεργοποίησης. Πρόκειται για τη συνάρτηση Heaviside, η οποία, για οποιαδήποτε τιμή του αθροίσματος v_k , μπορεί να πάρει μόνο δύο τιμές, μηδέν όταν $v_k \geq 0$ ή μονάδα όταν $v_k < 0$. Έτσι, μπορεί να χρησιμοποιηθεί για την ταξινόμηση στοιχείων σε δύο κλάσεις. Προϋπόθεση αποτελεί η ύπαρξη γραμμικής συνάρτησης τέτοιας, ώστε να χωρίζει σε δύο τμήματα ένα n -διάστατο χώρο (n ο αριθμός των σημάτων εισόδου x_n). Αυτό γίνεται ευκολότερα αντιληπτό για δύο σήματα εισόδου x , από το ακόλουθο σχήμα.



Σχήμα 1.3. Γραμμικός διαχωρισμός (α) και μη γραμμικός διαχωρισμός (β), ζεύγη τιμών x_1, x_2

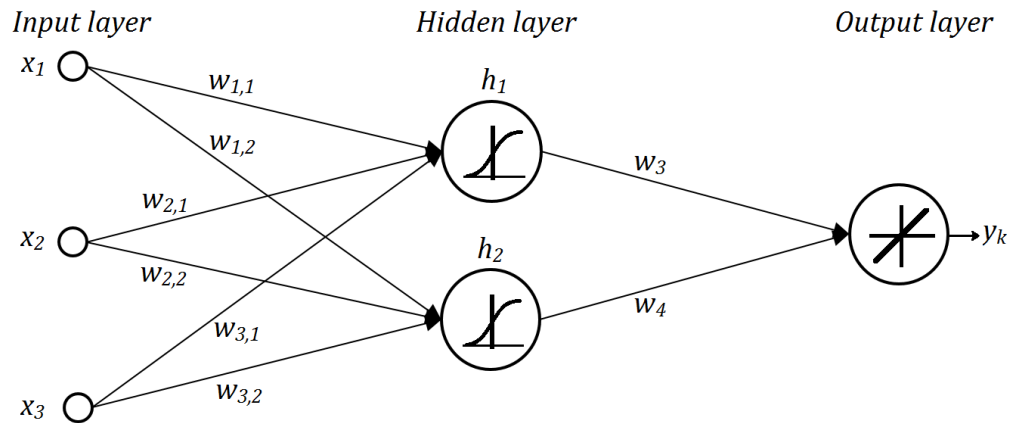
Στην περίπτωση (β) του σχήματος 1.3, δεν υπάρχει ευθεία γραμμή τέτοια, ώστε να μπορεί να χωρίσει τα στοιχεία σε δύο κλάσεις. Επιπλέον υπάρχουν εφαρμογές όπου δεν μπορούν να χρησιμοποιηθούν τέτοιοι νευρώνες. Τα προβλήματα αυτά θα μπορούσαν να αντιμετωπιστούν χρησιμοποιώντας ένα δίκτυο νευρώνων, ωστόσο δεν ήταν διαθέσιμο εκείνη την εποχή υπολογιστές με αρκετή υπολογιστική ισχύ για τη διεκπεραίωση της εκπαίδευσης, δηλαδή για τον υπολογισμό των συντελεστών βαρύτητας ενός νευρωνικού δικτύου.

Οι παραπάνω περιορισμοί, σε συνδυασμό με το γεγονός ότι η συνάρτηση Heaviside παρουσιάζει κρίσιμο σημείο για $v_k = 0$, όπου δεν είναι παραγωγίσιμη, οδήγησαν στη δημιουργία του σιγμοειδούς νευρώνα. Ο σιγμοειδής νευρώνας αποτελεί εξέλιξη του perceptron, διαφέροντας από αυτόν μόνο ως προς τη συνάρτηση ενεργοποίησης, η οποία σε αυτή την περίπτωση είναι σιγμοειδής. Αναλυτικά:

$$\varphi(v_k) = \frac{1}{1 + e^{-v_k}} \tag{1.12}$$

Βασικά χαρακτηριστικά της παραπάνω συνάρτησης αποτελούν ο περιορισμός του σήματος εξόδου στο διάστημα $[0,1]$, η πλήρης παραγωγισιμότητά της για κάθε τιμή v_k καθώς και το γεγονός ότι μπορεί να λάβει οποιαδήποτε τιμή στο διάστημα $[0,1]$. Από αυτά τα χαρακτηριστικά, το δεύτερο αποτελεί απαραίτητο στοιχείο για την εφαρμογή των αλγορίθμων εκπαίδευσης, όπως αναφέρεται παρακάτω, ενώ το τρίτο καθιστά τη χρήση των σιγμοειδών νευρώνων κατάλληλη για προβλήματα παλινδρόμησης.

Οι νευρώνες, όπως προαναφέρθηκε, αποτελούν το βασικό κομμάτι κάθε νευρωνικού δικτύου. Ένα νευρωνικό δίκτυο αποτελείται, στην απλούστερη μορφή του, από τρία επίπεδα (τα οποία αναφέρονται στη βιβλιογραφία ως layers). Ένα απλό νευρωνικό δίκτυο φαίνεται στο ακόλουθο σχήμα.

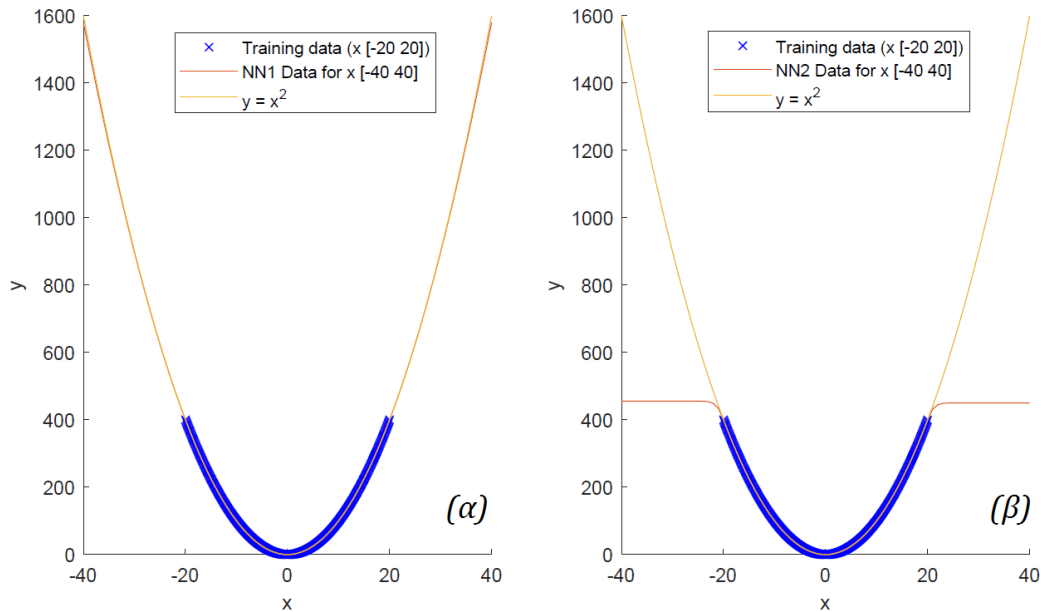


Σχήμα 1.4. Σχεδιάγραμμα τεχνητού νευρωνικού δικτύου.

Στο πρώτο επίπεδο (που ονομάζεται επίπεδο εισόδου - input layer) πραγματοποιείται η εισαγωγή των σημάτων x_i . Κάθε σήμα εισόδου x_i συνδέεται με κάθε νευρώνα του δεύτερου επιπέδου (που αναφέρεται ως εσωτερικό ή κρυφό επίπεδο - hidden layer) μέσω συνάψεων με συντελεστές βαρύτητας $w_{i,j}$. Σε κάθε νευρώνα του εσωτερικού επιπέδου εφαρμόζεται η σχέση (1.11) (δηλαδή πολλαπλασιασμός όλων των εισερχομένων στο νευρώνα σημάτων με τους αντίστοιχους συντελεστές βαρύτητας, άθροιση και χρήση του αθροίσματος ως όρισμα της σιγμοειδούς συνάρτησης ενεργοποίησης). Η έξοδος κάθε νευρώνα του hidden layer αποτελεί την είσοδο των νευρώνων του τρίτου και τελευταίου επιπέδου (που ονομάζεται επίπεδο εξόδου - output layer), όπου ακολουθείται η ίδια διαδικασία. Τα αποτελέσματα y_k που προκύπτουν από τους νευρώνες εξόδου αποτελούν σήματα εξόδου του νευρωνικού δικτύου. Να σημειωθεί ότι, σε κάθε νευρώνα (υπάρχουν τρεις στο σχήμα 1.4) αντιστοιχεί πάντα ένας συντελεστής βαρύτητας w_0 (bias), αν και δεν είναι ορατοί στο παραπάνω σχήμα.

Ο αριθμός των σημάτων εισόδου x_i ενός νευρωνικού δικτύου καθώς και των νευρώνων του επιπέδου εξόδου εξαρτάται από τη φύση του προβλήματος για την επίλυση του οποίου χρησιμοποιείται. Έτσι, για παράδειγμα, σε ένα πρόβλημα προσέγγισης της συνάρτησης $y = x^2$ (πρόβλημα παλινδρόμησης) θα υπάρχει μόνο ένα σήμα εισόδου x και ένας νευρώνας εξόδου ώστε να παράγονται από το νευρωνικό δίκτυο τιμές της μεταβλητής y .

Ο αριθμός εσωτερικών επιπέδων του δικτύου καθώς και των νευρώνων εκάστου επιπέδου, εξαρτάται επίσης από τη φύση του προβλήματος. Έχει θεωρητικά αποδειχθεί, ότι με χρήση ενός μόνο κρυφού επιπέδου, στους νευρώνες του οποίου χρησιμοποιούνται σιγμοειδείς συναρτήσεις ενεργοποίησης, μπορεί να προσεγγιστεί οποιαδήποτε συνεχής συνάρτηση (Cybenko 1989 – γνωστό και ως universal approximation theorem). Στην πράξη, απαιτείται ο σχεδιασμός της αρχιτεκτονικής των νευρωνικών δικτύων ώστε να επιτυγχάνεται ισορροπία μεταξύ της επιθυμητής ακρίβειας και του υπολογιστικού κόστους που απαιτείται για την εκπαίδευσή τους. Επίσης, η χρήση μεγάλου αριθμού νευρώνων (αναλογικά με το πόσοι απαιτούνται για να φτιαχτεί μία ικανοποιητική αρχιτεκτονική) μπορεί να δημιουργήσει πρόβλημα στην ικανότητα γενίκευσης του δικτύου (overfitting). Γενίκευση ονομάζεται η ικανότητα ενός νευρωνικού δικτύου να χρησιμοποιείται για προβλέψεις σε περιοχές δεδομένων όπου δεν χρησιμοποιήθηκαν για εκπαίδευση. Αυτό γίνεται καλύτερα αντιληπτό με το ακόλουθο διάγραμμα:



Διάγραμμα 1.2 (α), (β). Στο διάγραμμα (β) παρουσιάζονται αποτελέσματα που προέρχονται από νευρωνικό δίκτυο με μηδενική ικανότητα γενίκευσης.

Στο διάγραμμα 1.2 παρουσιάζεται, τόσο στο (α) όσο και στο (β), η συνάρτηση $y = x^2$ για ένα εύρος τιμών του x από -40 έως 40 (κίτρινη γραμμή). Έχει πραγματοποιηθεί εκπαίδευση δύο νευρωνικών δικτύων, NN1 (α) και NN2 (β) χρησιμοποιώντας ως δεδομένα 400 τιμές του x στο διάστημα $[-20, 20]$ (μπλε σημεία). Το NN1 προσεγγίζει τη συνάρτηση μέσω ενός νευρωνικού δικτύου με 3 κρυφούς νευρώνες και χρήση $\varphi = \tanh$ ως συνάρτηση ενεργοποίησης ενώ το NN2 αποτελείται από 20 κρυφούς νευρώνες με την ίδια συνάρτηση ενεργοποίησης, ισομοιρασμένους σε δύο εσωτερικά επίπεδα. Αν και ο αριθμός των συντελεστών βαρύτητας είναι πολύ μεγαλύτερος στη δεύτερη περίπτωση (141 έναντι 10), η ικανότητα του δικτύου για πρόβλεψη σε περιοχές όπου δεν υπήρχαν δεδομένα κατά την εκπαίδευση ($[-40, -20]$ και $[20, 40]$) είναι μηδενική.

Ο αριθμός των συντελεστών βαρύτητας N_w ο οποίος αναφέρθηκε στην προηγούμενη παράγραφο εξαρτάται από τον αριθμό των νευρώνων ενός δικτύου, τον αριθμό των εσωτερικών επιπέδων καθώς και το πλήθος των σημάτων εισόδου και εξόδου. Συγκεκριμένα, για ένα νευρωνικό δίκτυο που αποτελείται από I σήματα εισόδου, O σήματα εξόδου (και συνεπώς νευρώνες εξόδου) και n εσωτερικά επίπεδα με H_n νευρώνες στο κάθε ένα, υπολογίζεται (συμπεριλαμβάνοντας τον αριθμό των συντελεστών w_o), από την ακόλουθη σχέση:

$$N_w = I \cdot H_1 + H_1 H_2 + \dots + H_{n-1} H_n + (H_n + 1) \cdot O + \sum_{i=1}^n H_n \quad (1.13)$$

Η σχέση (1.13) δηλώνει πρακτικά ότι ο συνολικός αριθμός των συντελεστών βαρύτητας ισούται με τον αριθμό των συνάψεων και των νευρώνων αφού σε κάθε σύναψη ανατίθεται ένας συντελεστής βαρύτητας $w_{i,j}$ και σε κάθε νευρώνα ένας συντελεστής w_o . Ο αριθμός N_w αποτελεί ένα ενδεικτικό μέτρο του χρόνου που απαιτείται για την εκπαίδευση ενός νευρωνικού δικτύου.

Το σχεδιασμό ενός νευρωνικού δικτύου ακολουθεί η διαδικασία εκπαίδευσης. Όπως αναφέρθηκε στην ενότητα 'Περί μηχανικής μάθησης', τα νευρωνικά δίκτυα χρησιμοποιούνται στο πλαίσιο της συγκεκριμένης διπλωματικής ως εργαλεία παλινδρόμησης, που αποτελεί μία μορφή 'εποπτευόμενης' μάθησης (supervised learning). Ο εκπαιδευτής είναι δηλαδή αυτός ο οποίος παρέχει στον αλγόριθμο εκπαίδευσης τις τιμές εκείνες του σήματος εξόδου που αντιστοιχούν σε κάθε τιμή του σήματος εισόδου. Η εκπαίδευση ενός νευρωνικού δικτύου είναι διαδικασία που ομοιάζει με αυτή που ακολουθείται για την κατασκευή ενός μοντέλου γραμμικής παλινδρόμησης. Όπως δηλαδή μία ευθεία που περιγράφεται από την εξίσωση $y = ax + b$, προσαρμόζεται σε ζεύγη τιμών (x, y) και αναζητούνται μοναδικές τιμές των παραμέτρων a και b ώστε να ελαχιστοποιείται μία συνάρτηση κόστους C , έτσι και ο αλγόριθμος εκπαίδευσης νευρωνικού δικτύου αναζητά τις τιμές εκείνες των παραμέτρων $w_{i,j}$ και w_o που ελαχιστοποιούν τη συνάρτηση κόστους. Στην περίπτωση της γραμμικής παλινδρόμησης ως συνάρτηση

κόστους ή σφάλματος χρησιμοποιείται το άθροισμα των τετραγώνων των διαφορών μεταξύ πραγματικής και προβλεπόμενης τιμής $y(x)$ για κάθε ένα από τα n ζεύγη (x_i, y_i) :

$$C = \sum_{i=1}^n (y_{pred} - y_i)^2 \quad (1.14)$$

όπου y_{pred} η προβλεπόμενη από το μοντέλο τιμή της μεταβλητής y . Όμοια, στην περίπτωση εκπαίδευσης νευρωνικού δικτύου χρησιμοποιείται η συνάρτηση σφάλματος:

$$E = \frac{1}{n} \sum_{i=1}^n (y_{pred} - y_i)^2 \quad (1.15)$$

όπου y_{pred} το προβλεπόμενο και y το επιθυμητό διάνυσμα των σημάτων εξόδου αντίστοιχα και n ο αριθμός των συντελεστών βαρύτητας. Πυρήνα του αλγόριθμου εύρεσης των συντελεστών βαρύτητας w , που ελαχιστοποιούν τη συνάρτηση E και ο οποίος ονομάζεται αλγόριθμος *backpropagation*, αποτελεί η μέθοδος που ονομάζεται *gradient descent*. Η μέθοδος βασίζεται στον υπολογισμό του διανύσματος $grad(E)$, δηλαδή:

$$\vec{\nabla}E \equiv \left[\frac{\partial E}{\partial w_0}, \frac{\partial E}{\partial w_1}, \dots, \frac{\partial E}{\partial w_n} \right] \quad (1.16)$$

Το διάνυσμα αυτό εκφράζει την κατεύθυνση στην οποία η (κατά κατεύθυνση) παράγωγος της συνάρτησης σφάλματος λαμβάνει τη μέγιστη τιμή της. Ομοίως, το διάνυσμα $-\vec{\nabla}E$ δηλώνει την κατεύθυνση στην οποία η παράγωγος της συνάρτησης E λαμβάνει την ελάχιστη (αρνητική) τιμή της, κατεύθυνση προς την οποία παρουσιάζει τη μέγιστη πτώση. Ξεκινώντας λοιπόν από τυχαίες αρχικές τιμές του διανύσματος $[w_0 \ w_1 \ \dots \ w_n]$, ο αλγόριθμος εκπαίδευσης πορεύεται προς την κατεύθυνση της μέγιστης πτώσης της συνάρτησης E , μεταβάλλοντας τους συντελεστές w , ως εξής:

$$\begin{bmatrix} w_0^{(k+1)} \\ w_1^{(k+1)} \\ \vdots \\ w_n^{(k+1)} \end{bmatrix} = \begin{bmatrix} w_0^{(k)} \\ w_1^{(k)} \\ \vdots \\ w_n^{(k)} \end{bmatrix} - \eta \begin{bmatrix} \frac{\partial E}{\partial w_0}^{(k)} \\ \frac{\partial E}{\partial w_1}^{(k)} \\ \vdots \\ \frac{\partial E}{\partial w_n}^{(k)} \end{bmatrix} \quad (1.17)$$

όπου η , μία θετική σταθερά που ονομάζεται ρυθμός εκμάθησης (learning rate) και k , ο αριθμός της επανάληψης. Βεβαίως, προκειμένου να μπορούν να υπολογιστούν οι παράγωγοι $\frac{\partial E}{\partial w} = \frac{\partial E}{\partial y} \frac{\partial y}{\partial w}$, όπως προκύπτει από τον κανόνα της αλυσίδας, οφείλει η συνάρτηση y , δηλαδή το σήμα εξόδου ενός νευρώνα που ισούται με $y = \varphi(v_k)$, όπου $v_k = \sum(w \cdot x)$, να είναι παραγωγίσιμη.

Ένα πρόβλημα που προκύπτει κατά την εφαρμογή του αλγόριθμου εκπαίδευσης *backpropagation* είναι η διακοπή της εκπαίδευσης με σύγκλιση σε τοπικό ελάχιστο της συνάρτησης σφάλματος. Ένα νευρωνικό δίκτυο με N_w συντελεστές βαρύτητας θα έχει συνάρτηση σφάλματος N_w διαστάσεων καθιστώντας πρακτικά αδύνατη την εύρεση του ολικού ελαχίστου. Το πρόβλημα αυτό δεν αποδεικνύεται στην πράξη πολύ σοβαρό (Mitchell, 1997). Επειδή ο αλγόριθμος ξεκινά από τυχαίες αρχικές τιμές του διανύσματος w όπως προαναφέρθηκε, το μερικό ελάχιστο στο οποίο συγκλίνει μπορεί να είναι διαφορετικό για κάθε εκπαίδευση. Έτσι, στα πλαίσια της συγκεκριμένης εργασίας, σε όποιο σημείο αναφέρεται ότι πραγματοποιήθηκε εκπαίδευση νευρωνικού δικτύου, εννοείται η εκπαίδευση ενός αριθμού νευρωνικών δικτύων (από τρία έως δέκα), με ίδια χαρακτηριστικά και επιλογή του καλύτερου ως προς το σφάλμα αλλά και την ικανότητα γενίκευσης.

ΚΕΦΑΛΑΙΟ ΔΕΥΤΕΡΟ

Μεθοδολογική προσέγγιση και εφαρμογή σε περιβάλλον Matlab

1. Συνοπτική περιγραφή της μεθοδολογίας

Στόχος της παρούσης εργασίας είναι, όπως προαναφέρθηκε, η εφαρμογή μεθόδων μηχανικής μάθησης, για την επιλογή παραμέτρων και την ανάπτυξη μοντέλων, που θα προσεγγίζουν το δεξί σκέλος διαφορικής εξίσωσης και ο συνδυασμός των μεθόδων μηχανικής μάθησης με κλασσικές μεθόδους αριθμητικής ανάλυσης. Προς επίτευξη του στόχου αυτού ακολουθείται η μεθοδολογική προσέγγιση που περιγράφεται σε αυτό το κεφάλαιο το οποίο αποτελείται από τρεις ενότητες.

Στην πρώτη ενότητα (2) χρησιμοποιείται μία απλή μερική διαφορική εξίσωση για την παραγωγή των απαραίτητων δεδομένων, που αξιοποιούνται εν συνεχεία κατά την εφαρμογή διεργασίας Gauss και την εκπαίδευση νευρωνικών δικτύων. Μέσω της διεργασίας Gauss επιλέγονται μεταβλητές οι οποίες επηρεάζουν σημαντικά το δεξί σκέλος της διαφορικής εξίσωσης, θεωρώντας αυτή μη γνωστή. Έπειτα κατασκευάζονται μοντέλα νευρωνικού δικτύου τα οποία χρησιμοποιούνται ως μοντέλα παλινδρόμησης, υπολογίζοντας, για κάθε τιμή των επιλεγμένων μεταβλητών εισόδου, την τιμή του αριστερού σκέλους της διαφορικής εξίσωσης, ήτοι, του χρονικού διαφορικού της εξαρτημένης μεταβλητής ($\partial y/\partial t$). Με τα μοντέλα αυτά προσεγγίζεται η λύση της διαφορικής εξίσωσης σε μεταβατική κατάσταση χρησιμοποιώντας ode solvers και μάλιστα σε διαφορετικό διάστημα της χωρικής μεταβλητής (x) και για μεγαλύτερους χρόνους σε σχέση με τα δεδομένα που χρησιμοποιήθηκαν για την εκπαίδευση. Τα αποτελέσματα συγκρίνονται με την 'πραγματική λύση', όπως αυτή παράγεται με τη μέθοδο των πεπερασμένων στοιχείων.

Στην δεύτερη ενότητα (3.1) γίνεται χρήση ενός συστήματος δύο αδιάστατων μερικών διαφορικών εξισώσεων, που περιγράφει αζονική διασπορά σε αυλωτό αντιδραστήρα με μη αντιστρεπτή κινητική, πρώτης τάξης (Jensen & Ray, 1982). Αρχικά, όπως και στην ενότητα (2), παράγονται δεδομένα από επίλυση του συστήματος των διαφορικών εξισώσεων. Με τα δεδομένα αυτά πραγματοποιείται εκπαίδευση μοντέλου νευρωνικού δικτύου το οποίο παράγει ως μεταβλητή εξόδου διάνυσμα με τα μερικά χρονικά διαφορικά των μεταβλητών του συστήματος. Τα διανύσματα αυτά χρησιμοποιούνται, ομοίως με την ενότητα (2), για την εύρεση της μεταβατικής κατάστασης του συστήματος μέσω ode solvers. Δημιουργείται επίσης μοντέλο νευρωνικού δικτύου με χρήση των δεδομένων ενός μόνο κλάδου λύσεων του συστήματος (όπως θα δειχθεί παρακάτω ανάλογα με την τιμή μίας παραμέτρου, το σύστημα μπορεί να έχει λύση σε έναν από δύο κλάδους ή να εμφανίζει πολλαπλότητα). Στόχος είναι να εξακριβωθεί εάν μπορεί να γίνει πρόβλεψη που κλάδου λύσεων, δεδομένα του οποίου δεν χρησιμοποιήθηκαν κατά τη διαδικασία εκπαίδευσης. Πραγματοποιείται σύγκριση με αποτελέσματα που προέκυψαν μέσω της μεθόδου πεπερασμένων στοιχείων.

Στην τρίτη ενότητα (3.2) εξετάζεται το καλύτερο από τα μοντέλα που παρήχθησαν με τη διαδικασία της ενότητας (3.1) για την επίλυση του προβλήματος εύρεσης της μόνιμης κατάστασης του συστήματος. Προς τούτο αξιοποιείται μία μέθοδος Newton-Krylov και συγκεκριμένα η Newton-GMRES.

Στα πλαίσια του συγκεκριμένου κεφαλαίου γίνεται επίσης αναφορά στον τρόπο εφαρμογής των προαναφερθεισών μεθόδων σε περιβάλλον Matlab.

2. Προσέγγιση συνάρτησης με επιλογή κατάλληλων παραμέτρων

Οι μέθοδοι μηχανικής μάθησης εκ φύσεως απαιτούν την ύπαρξη μεγάλου όγκου διαθέσιμων δεδομένων για την εφαρμογή τους. Ως εκ τούτου, η ακόλουθη διαδικασία αποσκοπεί στην παραγωγή δεδομένων, που αξιοποιούνται για την εφαρμογή της υπό ανάπτυξη μεθοδολογίας και προς κατάδειξη της χρησιμότητάς της.

Έστω η μερική διαφορική εξίσωση που δίνεται, στη γενική της μορφή, ως ακολούθως και περιγράφει μονοδιάστατα προβλήματα μεταβατικής κατάστασης:

$$\frac{\partial y(x, t)}{\partial t} = f\left(x, y, \frac{\partial y(x, t)}{\partial x}, \frac{\partial^2 y(x, t)}{\partial x^2}\right) \quad (2.1)$$

Από την παραπάνω εξίσωση θεωρώντας:

$$f\left(x, y, \frac{\partial y(x, t)}{\partial x}, \frac{\partial^2 y(x, t)}{\partial x^2}\right) = \frac{\partial^2 y}{\partial x^2} + y^2, x \in (0,1) \quad (2.2)$$

προκύπτει η μερική διαφορική εξίσωση:

$$\frac{\partial y}{\partial t} = \frac{\partial^2 y}{\partial x^2} + y^2, x \in (0,1) \quad (2.3)$$

Η εξίσωση (2.3) είναι μία μη γραμμική, δεύτερης τάξης, μερική διαφορική εξίσωση με ανεξάρτητες μεταβλητές το χρόνο (t) και μία χωρική μεταβλητή (x). Η επίλυσή της μπορεί να πραγματοποιηθεί με οποιαδήποτε από τις υπάρχουσες μεθόδους (όπως για παράδειγμα με τη μέθοδο των γραμμών). Εν προκειμένω, εφαρμόζεται η μέθοδος των πεπερασμένων στοιχείων, αξιοποιώντας το λογισμικό COMSOL Multiphysics (έκδοση 5.2). Αρχικά απαιτείται η μετατροπή της υπό εξέταση εξίσωσης σε μορφή συμβατή με το λογισμικό. Μέσω της μεθόδου Galerkin παράγεται η ασθενής μορφή (weak formulation) της μερικής διαφορικής εξίσωσης (2.3). Έπειτα απαιτείται η εισαγωγή αρχικών συνθηκών, των τιμών δηλαδή που παίρνει η εξαρτημένη μεταβλητή y , τη στιγμή t_0 ($t_0 = 0$) για κάθε τιμή του x . Χρησιμοποιήθηκαν δώδεκα αρχικές συνθήκες, οι εξισώσεις των οποίων παρατίθενται στον παρακάτω πίνακα.

Όνομα αρχικής συνθήκης/ Περιγραφή	Εξίσωση
GP_1 Κανονική κατανομή	$y(x) = \frac{1}{0,1 \cdot \sqrt{2\pi}} \cdot \exp\left(-\frac{(x-0,5)^2}{2 \cdot 0,1^2}\right)$
GP_2 Κανονική κατανομή	$y(x) = 3 \cdot \exp\left(-\frac{(2x-1,2)^2}{0,2^2}\right)$
SIN_1 Ημιτονοειδής συνάρτηση	$y(x) = \sin(2\pi x)$
PULSE_1 Ορθογώνιος παλμός	$y(x) = \begin{cases} 0, & 0 < x < 0,1 \\ 1, & 0,1 < x < 0,9 \\ 0, & 0,9 < x < 1 \end{cases}$
TRIG_1 Τριγωνικός παλμός	$y(x) = \begin{cases} 0, & 0,0 < x < 0,1 \\ 2,5(x-0,1), & 0,1 < x < 0,5 \\ -2,5(x-1), & 0,5 < x < 0,9 \\ 0, & 0,9 < x < 1,0 \end{cases}$
SIGM_1 Σιγμοειδής συνάρτηση	$y(x) = \frac{10}{10 + 1 \cdot \exp(10(x-0,25))}$
SIGM_2 Σιγμοειδής συνάρτηση	$y(x) = 4 \cdot \exp\left(-\frac{(0,5x-0,5)^2}{0,2^2}\right)$
GP_SIN_1 Γινόμενο συναρτήσεων	$y(x) = \exp\left(-\frac{(x-0,5)^2}{0,1^2}\right) \cdot \sin(2\pi x)$
GP_SIN_2 Γινόμενο συναρτήσεων	$y(x) = \exp\left(-\frac{(x-0,4)^2}{0,1^2}\right) \cdot \sin(2\pi x)$
GP_SIGM_1 Γινόμενο συναρτήσεων	$y(x) = \exp(-(x-0,5)^2) \cdot \left(\frac{10}{10 + 1 \cdot \exp(10(x-0,25))}\right)$
POLY_1 Πολυωνυμική συνάρτηση	$y(x) = (x-0,5)^3$
POLY_2 Πολυωνυμική συνάρτηση	$y(x) = x^{0,7}$

Σημειώνεται ότι στην περίπτωση των αρχικών συνθηκών PULSE_1 και TRIG_1 πραγματοποιήθηκε λείανση των γωνιών ώστε να εξασφαλίζεται πλήρης παραγωγισιμότητα.

Οι συνοριακές συνθήκες οι οποίες χρησιμοποιήθηκαν ήταν, σε κάθε περίπτωση, πρώτου είδους (Dirichlet) και ίσες με την τιμή της εκάστοτε συνάρτησης $y(x)$ για $x_1 = 0$ και $x_2 = 1$ προκειμένου να μην εμφανιστούν απότομες μεταβολές οι οποίες να οφείλονται στις συνοριακές συνθήκες. Για κάθε αναγραφόμενη αρχική συνθήκη παρήχθησαν λύσεις της διαφορικής εξίσωσης (2.3) στο διάστημα $x \in [0,1]$ με βάση το πρωτόκολλο που παρουσιάζεται στον ακόλουθο πίνακα:

Πίνακας 2.2

Χρόνοι και πυκνότητες πλέγματος που χρησιμοποιήθηκαν για παραγωγή δεδομένων μέσω Comsol.

Αρχικός χρόνος	Τελικός χρόνος	Βήμα	Πυκνότητα πλέγματος	Αρχικές συνθήκες
0,001	0,003	0,001	Extremely Fine	Όλες οι αναγραφόμενες στον πίνακα 2.1
0,01	0,1	0,01	Finer	Οι αναγραφόμενες στον πίνακα 2.1, πλην GP_2, SIGM_2, POLY_1, POLY_2
0,1	1	0,1	Finer	

Η πυκνότητα πλέγματος 'extremely fine' αντιστοιχεί σε χωρισμό του διαστήματος x σε 100 τμήματα η δε πυκνότητα πλέγματος 'finer' σε 28 τμήματα.

Τα δεδομένα που εξήχθησαν είχαν τη μορφή τιμών του διανύσματος $[y_t \ y_x \ y_{xx}]$. Η διαδικασία παραγωγής δεδομένων από το Comsol Multiphysics, από τη δημιουργία μοντέλου μέχρι την εξαγωγή δεδομένων, περιγράφεται αναλυτικά στο παράρτημα (Α). Συνολικά παρήχθησαν 8.000 τετράδες δεδομένων $[y_t \ y_x \ y_{xx}]$. Σε τμήματα αυτών των δεδομένων (και συγκεκριμένα σε αυτά που αντιστοιχούσαν στις αρχικές συνθήκες GP_1, SIN_1, PULSE_1, TRIG_1) εφαρμόστηκαν διεργασίες Gauss χρησιμοποιώντας ως συνάρτηση kernel τη 'Radial Basis Function'. Κρίθηκε δηλαδή σκόπιμο να εφαρμοστούν διεργασίες Gauss τέσσερις φορές σε μικρά τμήματα δεδομένων παρά στο σύνολό τους, λόγω μεγάλου υπολογιστικού κόστους (υπενθυμίζεται ότι ο χρόνος εκπαίδευσης διεργασίας Gauss αποτελεί συνάρτηση του κύβου του αριθμού χρησιμοποιούμενων δεδομένων: $t = O(n^3)$) καθώς και διότι ο αριθμός των χρησιμοποιούμενων δεδομένων αποδείχθηκε αρκετός για την εξαγωγή συμπεράσματος για τις χρήσιμες μεταβλητές.

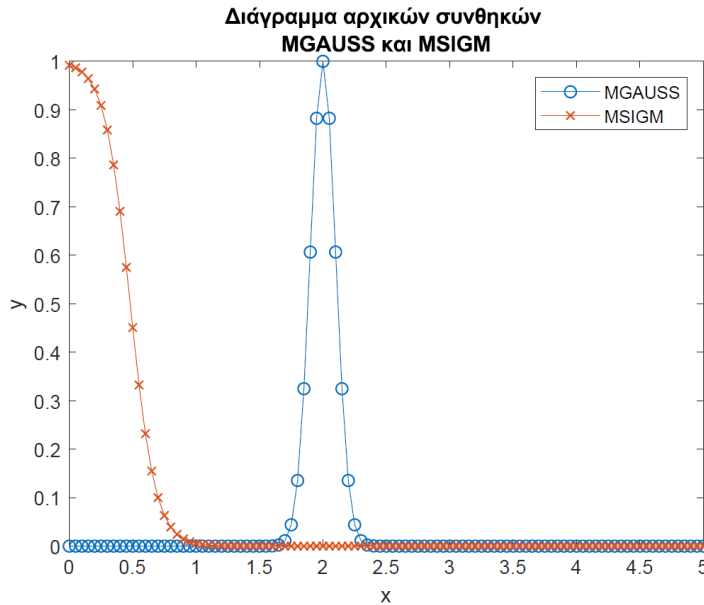
Έπειτα, εκπαιδεύτηκε μοντέλο νευρωνικού δικτύου χρησιμοποιώντας τα δεδομένα που παρήχθησαν με την προηγούμενη διαδικασία προκειμένου να χρησιμοποιηθεί ως συνάρτηση προσέγγισης του δεξιού σκέλους της διαφορικής εξίσωσης (2.3) και εύρεσης της μεταβατικής κατάστασης. Τα χαρακτηριστικά του μοντέλου παρατίθενται στον παρακάτω πίνακα:

Πίνακας 2.3	Χαρακτηριστικά εκπαιδευμένου νευρωνικού δικτύου
Όνομα νευρωνικού δικτύου	NET7000K
Αριθμός νευρώνων (ένα εσωτερικό επίπεδο)	15
Συνάρτηση μεταφοράς εσωτερικών νευρώνων	$\varphi = \tanh(v_k)$
Συνολικός αριθμός δεδομένων που χρησιμοποιήθηκαν για εκπαίδευση	7.000
Πλήθος διαθέσιμων δεδομένων για εκπαίδευση	8.600

Το μοντέλο εφαρμόστηκε για την προσέγγιση λύσεων της (2.3) μέσω ode solver και εξετάστηκε η προσεγγιστική ικανότητά του, για διαφορετικές αρχικές και συνοριακές συνθήκες. Συγκεκριμένα πραγματοποιήθηκαν τέσσερις δοκιμές, με τα χαρακτηριστικά του παρακάτω πίνακα:

Πίνακας 2.4	Δοκιμές για εύρεση λύσεων εξίσωσης 2.3 μέσω εφαρμογής NN και ode solver			
Όνομα δοκιμής	Π1	Π2	Π3	Π4
Αρχική συνθήκη	MGAUSS	MGAUSS	MGAUSS	MSIGM
Συνοριακές συνθήκες	Dirichlet	Neumann	Robin	Dirichlet
	$y _{x=0} = 0$ $y _{x=5} = 0$	$(dy/dx) _{x=0} = c$ $(dy/dx) _{x=5} = c$ $c = 0,1$	$(dy/dx) _{x=0} = c_1 \cdot (y _{x=0} + c_2)$ $(dy/dx) _{x=5} = c_3 \cdot (y _{x=5} + c_4)$ $c_1 = 1, c_2 = 0,05$ $c_3 = -0,1, c_4 = 2$	$y _{x=0} = 1$ $y _{x=a} = 0,5$
Χρονικό διάστημα επίλυσης	[0,2]	[0,1]	[0,1]	[0,1]

Οι αρχικές συνθήκες MGAUSS και MSIGM που αναφέρονται στον πίνακα 2.4 παρουσιάζονται στο ακόλουθο διάγραμμα (2.1):



Διάγραμμα 2.1. Αρχικές συνθήκες MGAUSS και MSIGM που χρησιμοποιήθηκαν για την εφαρμογή του μοντέλου NN μέσω ode solver.

MGAUSS:

$$y = \exp\left(-\frac{(x-2)^2}{2 \cdot 0,1^2}\right)$$

MSIGM:

$$y = \frac{10}{(10 + \exp(10 \cdot (x - 0.25)))}$$

3.1 Πρόβλημα εύρεσης μεταβατικής κατάστασης συστήματος μερικών διαφορικών εξισώσεων

Έστω το σύστημα μερικών διαφορικών εξισώσεων το οποίο δίνεται (Jensen & Ray, 1982) ως εξής:

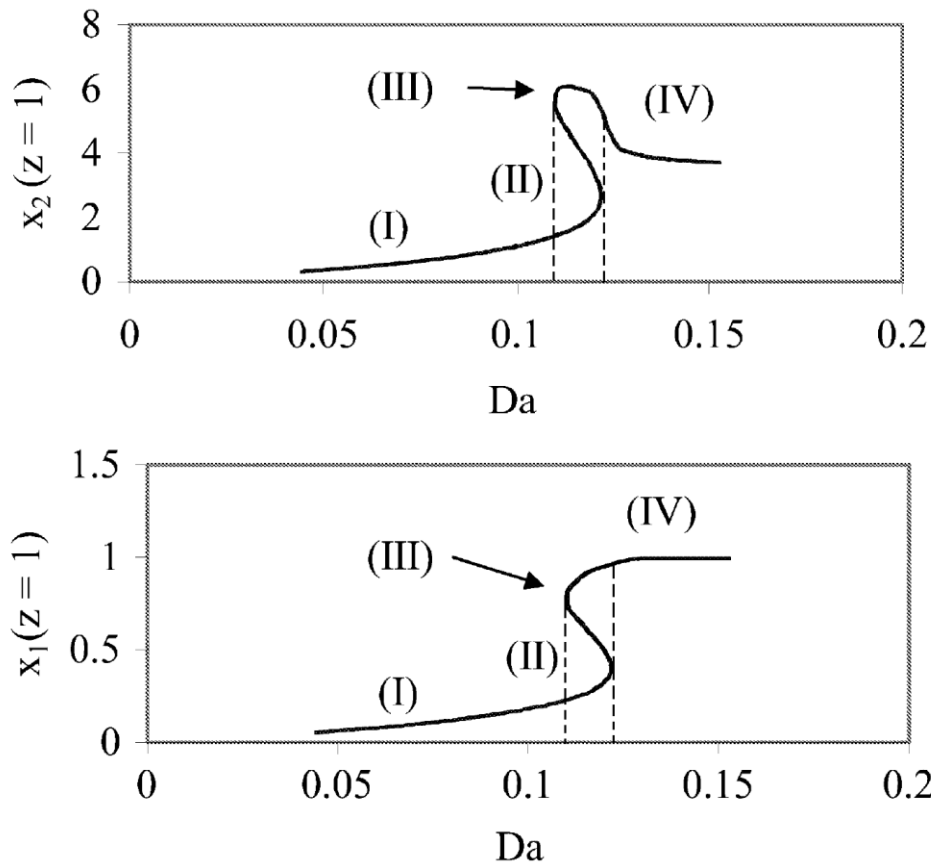
$$\frac{\partial x_1}{\partial t} = \frac{1}{Pe_1} \frac{\partial^2 x_1}{\partial z^2} - \frac{\partial x_1}{\partial z} + Da(1 - x_1) \exp\left(\frac{x_2}{1 + \frac{x_2}{\gamma}}\right) \quad (2.4, \alpha)$$

$$\frac{\partial x_2}{\partial t} = \frac{1}{Pe_2} \frac{\partial^2 x_2}{\partial z^2} - \frac{\partial x_2}{\partial z} - \beta \cdot x_2 + B \cdot Da(1 - x_1) \exp\left(\frac{x_2}{1 + \frac{x_2}{\gamma}}\right) \quad (2.4, \beta)$$

Στο παραπάνω σύστημα z και t είναι οι ανεξάρτητες και x_1, x_2 οι εξαρτημένες μεταβλητές, με x_1 τη μετατροπή εξώθερμης χημικής αντίδρασης με μη αντιστρεπτή κινητική πρώτης τάξης και x_2 την αδιάστατη θερμοκρασία αντίστοιχα. Με Da συμβολίζεται ο αριθμός Dahmköhler, Pe , ο αριθμός Péclet, β , ένας αδιάστατος συντελεστής μεταφοράς θερμότητας, B , αδιάστατη αδιαβατική αύξηση θερμοκρασίας και γ , αδιάστατη ενέργεια ενεργοποίησης. Στα πλαίσια της παρούσης εργασίας χρησιμοποιήθηκαν για τις παραπάνω παραμέτρους οι ακόλουθες τιμές (Koronaki, Boudounis & Kevrekidis, 2003):

Πίνακας 2.5		Τιμές αδιάστατων αριθμών συστήματος (2.4)	
$Pe_1 = Pe_2 = Pe = 5$	$\beta = 1,5$	$B = 12$	$\gamma = 20$

Ο αριθμός Dahmköhler, για τον οποίο δεν δίνεται τιμή στον παραπάνω πίνακα, αποτελεί παράμετρο του συστήματος εξισώσεων (2.4,α) και (2.4,β). Αυτό, ανάλογα με την τιμή του αριθμού Da , μπορεί να έχει ευσταθείς ή ασταθείς λύσεις. Στη δεύτερη περίπτωση, οι λύσεις δεν είναι δυνατόν να παρατηρηθούν σε πείραμα καθώς ελάχιστη μεταβολή προκαλεί μετάπτωση σε μία από δύο ευσταθείς καταστάσεις. Για τις συγκεκριμένες τιμές παραμέτρων του πίνακα 2.4, υπάρχει επίσης μία τιμή του αριθμού Da για την οποία εμφανίζεται περιοδική λύση. Στο ακόλουθο διάγραμμα φαίνονται οι τιμές των μεταβλητών x_1, x_2 στη μόνιμη κατάσταση ως προς την τιμή της παραμέτρου:



Διάγραμμα 2.2. Τιμές μεταβλητών x_1 και x_2 στη μόνιμη κατάσταση, για διαφορετικές τιμές της παραμέτρου Da . (I), (IV): Περιοχές με σταθερή μόνιμη κατάσταση, (II)- εντός διακεκομμένων γραμμών): Περιοχή πολλαπλότητας λύσεων, (III): Σημείο Hopf. Πηγή διαγράμματος: (Koronaki, Boudouvis & Kevrekidis, 2003)

Το σύστημα επιλύθηκε με τη μέθοδο των πεπερασμένων στοιχείων μέσω του Comsol Multiphysics. Οι εν λόγω εξισώσεις εισήχθησαν στο λογισμικό στη γενική μορφή (general form) και η επίλυσή τους πραγματοποιήθηκε στο διάστημα (interval) $[0,1]$, με χρήση πλέγματος (mesh) τέτοιου ώστε να χωρίζει το διάστημα z σε εκατό τμήματα, μήκους 0,01 έκαστο. Οι συνοριακές συνθήκες που επιβλήθηκαν ήταν οι εξής:

$$\left[\begin{array}{l} \frac{\partial x_1}{\partial t} = Pe \cdot x_1 \\ \frac{\partial x_2}{\partial t} = Pe \cdot x_2 \end{array} \right], z = 0 \text{ και } \left[\begin{array}{l} \frac{\partial x_1}{\partial t} = 0 \\ \frac{\partial x_2}{\partial t} = 0 \end{array} \right], z = 1$$

Οι χρονικές στιγμές, για τις οποίες ελήφθησαν δεδομένα για κάθε τιμή αριθμού Da και αρχική συνθήκη, επιλέχθηκαν ώστε να καλύπτουν ένα μικρό εύρος του χρονικού διαστήματος σε σχέση με αυτό που θα χρησιμοποιούνταν κατά την εφαρμογή του εκπαιδευμένου μοντέλου (ώστε να εξακριβωθεί το κατά πόσο το μοντέλο μαθαίνει τη σχέση μεταξύ των παραμέτρων) και αναγράφονται στον ακόλουθο πίνακα:

Πίνακας 2.6 Χρονικές στιγμές στις οποίες ελήφθησαν δεδομένα για κάθε τιμή αριθμού Da και αρχική συνθήκη

10^{-4}	$5 \cdot 10^{-4}$	10^{-3}	$5 \cdot 10^{-3}$	10^{-2}	$5 \cdot 10^{-2}$	10^{-1}	$2 \cdot 10^{-1}$	$5 \cdot 10^{-1}$
-----------	-------------------	-----------	-------------------	-----------	-------------------	-----------	-------------------	-------------------

Οι τιμές της παραμέτρου Da επιλέχθηκαν ώστε να συμπεριλαμβάνεται η περιοχή πολλαπλότητας λύσεων του συστήματος εξισώσεων (2.4,α και 2.4,β), δηλαδή την περιοχή (II) του διαγράμματος 2.2. Συγκεκριμένα χρησιμοποιήθηκαν οι τιμές που αναγράφονται στον παρακάτω πίνακα:

Πίνακας 2.7 Τιμές αριθμού Da για τις οποίες παρήχθησαν λύσεις του συστήματος

Αριθμός	Τιμή παραμέτρου Da	Αριθμός	Τιμή παραμέτρου Da
1.	0,080	7.	0,110
2.	0,085	8.	0,115
3.	0,090	9.	0,120
4.	0,095	10.	0,125
5.	0,100	11.	0,130
6.	0,105		

Ως αρχικές συνθήκες, χρησιμοποιήθηκαν είκοσι ζεύγη τιμών x_1, x_2 , διαφορετικά για κάθε τιμή του αριθμού Da . Οι αρχικές συνθήκες, σε αντίθεση με αυτές που χρησιμοποιήθηκαν στην ενότητα 2, Κεφ. II, ήταν γραμμικές, σταθερές κατά μήκος του διαστήματος z και οι τιμές τους φαίνονται στον πίνακα της επόμενης σελίδας.

Τα δεδομένα που προέκυψαν από την παραπάνω διαδικασία είχαν τη μορφή διανύσματος $[x_1 \ x_1 \ x_{1z} \ x_{1zz} \ x_2 \ x_{2l} \ x_{2z} \ x_{2zz}]$. Δεν επαναλήφθηκε η διαδικασία της ενότητας 2, Κεφ. II (εν προκειμένω θα μπορούσε με διεργασία Gauss ναδειχθεί ότι οι τιμές $x_{1,x2}$ ή $x_{2,x1}$ δεν επηρεάζουν το αποτέλεσμα αλλά η επανάληψη της διαδικασίας της προηγούμενης ενότητας δεν θα συνεισέφερε στην εξαγωγή επιπλέον συμπερασμάτων). Ο συνολικός αριθμός των παραγόμενων δεδομένων ανήλθε σε 198.000. Με αυτά εκπαιδεύτηκαν τα νευρωνικά δίκτυα με τα παρακάτω χαρακτηριστικά:

Πίνακας 2.8 Χαρακτηριστικά εκπαιδευμένων νευρωνικών δικτύων

Όνομα νευρωνικού δικτύου		NET3020_1_160	NET3510_2_87
Αριθμός νευρώνων	1° εσωτερικό επίπεδο	30	35
	2° εσωτερικό επίπεδο	20	10
Συνάρτηση μεταφοράς εσωτερικών νευρώνων		$\varphi = \tanh(v_k)$	$\varphi = \tanh(v_k)$
Συνολικός αριθμός δεδομένων που χρησιμοποιήθηκαν για εκπαίδευση		160.000	87.400
Πλήθος διαθέσιμων δεδομένων για εκπαίδευση		198.000	108.000
Τιμές παραμέτρου Da		0,080 – 0,130	0,080-0,105

Ο αριθμός των δεδομένων που χρησιμοποιήθηκαν για εκπαίδευση ήταν σε κάθε περίπτωση μικρότερος από το συνολικό αριθμό των διαθέσιμων δεδομένων. Συγκεκριμένα για το μοντέλο NET3020_1_160, αξιοποιήθηκε μόνο το 81 % του συνόλου των 198.000 δεδομένων, δηλαδή περίπου 160.000 σημεία. Ομοίως, για το νευρωνικό δίκτυο NET3510_2_87 χρησιμοποιήθηκε το 81 % των 108.000 που είχαν παραχθεί για τιμές του αριθμού Da στο διάστημα $[0,080 \ 0,105]$, διάστημα δηλαδή στο οποίο το σύστημα των εξισώσεων (2.4,α) και (2.4,β) εμφανίζει μόνιμη κατάσταση στον αριστερό κλάδο λύσεων (περιοχή I, διαγράμματος 2.2). Το ποσοστό των σημείων που χρησιμοποιήθηκαν (81 %) προέκυψε καθώς δεν χρησιμοποιήθηκαν δεδομένα που είχαν παραχθεί για z στα διαστήματα $[0 \ 0,09]$ και $[0,91 \ 1]$ προκειμένου να μην επηρεαστούν τα μοντέλα από την επίδραση των συνοριακών συνθηκών κατά την επίλυση του συστήματος μέσω πεπερασμένων στοιχείων (Comsol).

Η αρχιτεκτονική που επιλέχθηκε για κάθε ένα από τα παραπάνω δίκτυα βασίστηκε σε πλήθος δοκιμών με σταδιακή αύξηση του αριθμού των νευρώνων αλλά και δοκιμές με χρήση ενός ή δύο εσωτερικών επιπέδων με σκοπό τη χρήση του ελαχίστου αριθμού των συντελεστών βαρύτητας λόγω τεχνικών περιορισμών (ενδεικτικά αναφέρεται πως για την εκπαίδευση ενός δικτύου με $N_w = 1000$ απαιτούνταν περίπου 5 ώρες).

Τα τεχνητά νευρωνικά δίκτυα NET3020_1_160 και NET3510_2_87 αποτέλεσαν τις συναρτήσεις υπολογισμού των μερικών χρονικών διαφορικών $\partial x_1/\partial t$ και $\partial x_2/\partial t$ κατά τους προσεγγιστικούς υπολογισμούς λύσεων του συστήματος μέσω ode solver, για συγκεκριμένες αρχικές συνθήκες και τιμή της παραμέτρου Da . Οι αρχικές συνθήκες οι οποίες επιλέχθηκαν ήταν οι ακόλουθες:

Πίνακας 2.9 Τιμές αρχικών συνθηκών που επιλέχθηκαν για τη δοκιμή μοντέλων NN

Όνομα αρχικής συνθήκης	IC1	IC2	IC3	IC4	IC5	IC6	IC7
Μεταβλητή	x_1	0,1	0,9	0,5	0,5	0,3	0,7
	x_2	1	9	8	3	6	2

Οι παραπάνω αρχικές συνθήκες δεν μεταβάλλονταν κατά μήκος του διαστήματος z . Οι τιμές του αριθμού Da για τις οποίες έγιναν προσεγγίσεις (δοκιμές) ήταν $Da = 0,12$ και $Da = 0,118$, τιμές δηλαδή για τις οποίες το σύστημα μπορεί να έχει δύο ευσταθείς λύσεις.

Πίνακας 2.10 Αρχικές συνθήκες που χρησιμοποιήθηκαν για την παραγωγή δεδομένων με τη διαδικασία της ενότητας 3.1																			
1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.
Da = 0,080																			
0,1190	0,1386	0,1493	0,1626	0,2238	0,2551	0,2575	0,3404	0,4984	0,5060	0,5472	0,5853	0,6551	0,6797	0,6991	0,7513	0,8407	0,8909	0,9593	0,9597
4,3744	4,1691	1,8528	3,9112	4,9925	5,4331	3,1641	2,9395	7,9039	0,3019	7,0813	7,3063	6,3695	0,7897	2,0950	2,6829	5,4378	1,0924	5,7698	0,8541
Da = 0,085																			
0,0759	0,1966	0,2435	0,2511	0,2543	0,2858	0,3500	0,3517	0,3804	0,4733	0,5497	0,5678	0,5853	0,6160	0,7537	0,7572	0,8143	0,8308	0,9172	0,9293
5,2301	3,9534	6,2324	5,7203	7,2298	7,1274	2,6733	5,5900	1,5825	0,2443	5,9526	4,0002	3,8394	7,2378	4,8789	4,9413	6,8755	6,4439	4,6138	1,4634
Da = 0,090																			
0,0119	0,0540	0,1299	0,1622	0,1656	0,2630	0,3112	0,3371	0,4505	0,4694	0,5285	0,5308	0,5688	0,6020	0,6541	0,6892	0,7482	0,7792	0,7943	0,9340
1,9195	7,0921	0,2294	3,9192	1,3434	7,8294	5,7016	4,0038	3,7687	0,4770	5,4558	0,3394	2,5716	4,1732	2,7738	6,5452	6,5404	5,7795	1,1989	5,2768
Da = 0,095																			
0,0046	0,0782	0,0838	0,0944	0,1067	0,1524	0,2290	0,2599	0,3998	0,4314	0,4727	0,5383	0,7749	0,8001	0,8173	0,8258	0,8687	0,9133	0,9619	0,9961
4,1488	7,7838	5,1919	6,4026	3,6304	3,4591	6,6025	0,6678	1,0654	1,3871	3,1275	6,6510	6,4269	0,4838	3,1941	4,2150	3,3344	5,2549	5,0238	2,3359
Da = 0,100																			
0,0760	0,1233	0,1361	0,1450	0,1655	0,1818	0,1839	0,2399	0,2500	0,2638	0,3510	0,4018	0,4173	0,5132	0,5499	0,5797	0,6221	0,8530	0,8693	0,9106
3,4532	0,1239	7,8725	1,3373	0,8497	2,9793	1,5849	3,9175	2,7159	7,6130	7,3627	0,4214	5,9029	2,1530	3,3827	4,3830	7,5419	3,3420	7,8644	2,4116
Da = 0,105																			
0,0497	0,0598	0,0965	0,1112	0,1320	0,2348	0,2417	0,3377	0,3692	0,3897	0,4039	0,4893	0,4909	0,5752	0,7803	0,9001	0,9027	0,9421	0,9448	0,9561
5,6088	5,3307	4,3130	5,5848	5,3322	1,4251	1,0241	7,9926	1,3690	0,2608	4,4896	7,0549	5,3534	1,5235	2,9513	3,6858	7,8531	1,2512	6,8442	5,1581
Da = 0,110																			
0,0154	0,0430	0,0811	0,1690	0,1835	0,1890	0,2963	0,3532	0,3685	0,4509	0,5470	0,6256	0,6477	0,6491	0,6868	0,7317	0,7447	0,7802	0,8212	0,9294
3,0102	1,5274	3,4260	3,8562	0,9649	4,7161	1,8095	3,0770	4,6639	2,0144	2,3235	4,9367	2,1222	6,5950	7,8613	5,8420	2,7510	4,6726	0,8622	7,2505
Da = 0,115																			
0,2077	0,3063	0,3507	0,3787	0,4359	0,4468	0,4868	0,5085	0,5108	0,5328	0,5502	0,5870	0,6225	0,6443	0,7757	0,7948	0,8116	0,8176	0,8759	0,9390
7,0372	6,5421	2,0858	4,7549	0,1801	3,4021	2,5018	1,2919	1,4301	3,3831	0,7538	4,7882	3,7674	5,5676	5,5991	5,1082	2,2688	6,5504	2,5568	4,2469
Da = 0,120																			
0,1111	0,1707	0,1848	0,1948	0,2259	0,2277	0,2305	0,2581	0,3012	0,3111	0,4087	0,4302	0,4357	0,4389	0,4709	0,5949	0,8443	0,9049	0,9234	0,9797
5,2356	3,2610	6,5598	5,7469	7,7492	4,2507	2,6012	0,8450	4,8877	6,2304	3,3876	1,7266	2,1318	1,2293	2,2480	3,5207	4,2171	3,6594	7,0030	4,1444
Da = 0,125																			
0,0292	0,0855	0,1174	0,2217	0,2373	0,2622	0,2625	0,2967	0,3188	0,4242	0,4588	0,4886	0,5079	0,5785	0,6028	0,7112	0,7303	0,8010	0,9289	0,9631
7,5490	5,1017	7,6616	1,9257	5,4090	2,3125	5,3745	5,5611	0,5439	2,0383	1,7923	5,3427	6,7551	2,7557	6,2442	5,4027	0,0537	4,8174	3,0942	7,3279
Da = 0,130																			
0,9500	0,5700	0,0600	0,2400	0,3500	0,8200	0,0100	0,0400	0,1700	0,6500	0,7300	0,4500	0,2900	0,5200	0,3700	0,7800	0,6800	0,6200	0,9200	0,9900
0,7000	0,3000	0,6200	1,2200	1,8300	2,0100	2,8300	3,5400	3,7100	4,3100	4,6700	5,7300	5,9801	6,5700	6,9401	7,3100	7,6801	7,9602	8,2100	8,9000

3.2. Πρόβλημα εύρεσης μόνιμης κατάστασης συστήματος μερικών διαφορικών εξισώσεων

Το τεχνητό νευρωνικό δίκτυο NET3510_2_87, το οποίο εκπαιδεύτηκε με τη διαδικασία της ενότητας 3.1 του κεφαλαίου II, χρησιμοποιήθηκε για την προσέγγιση της μόνιμης κατάστασης του συστήματος των μερικών διαφορικών εξισώσεων (2.4,α) και (2.4,β) εφαρμόζοντας την αριθμητική μέθοδο Newton-GMRES.

Η μέθοδος Newton-GMRES αποτελεί υποκατηγορία των μεθόδων Newton-Krylov και χρησιμοποιείται όπως και η μέθοδος Newton-Raphson για επίλυση αλγεβρικής εξίσωσης με μία εξαρτημένη και ανεξάρτητη μεταβλητή. Έστω για παράδειγμα η αλγεβρική εξίσωση $y(x) = x^2$. Αναζητούνται τιμές του x τέτοιες, ώστε $y(x) = 0$. Όμοια, για την εύρεση της μόνιμης κατάστασης του συστήματος (στη μόνιμη κατάσταση ισχύει $\partial x/\partial t = 0$), που περιγράφεται από νευρωνικό δίκτυο, αναζητούνται τιμές \vec{x}_1 και \vec{x}_2 τέτοιες, ώστε:

$$\frac{\partial \vec{x}_1}{\partial t} = \vec{R}_1 = 0 \text{ και } \frac{\partial \vec{x}_2}{\partial t} = \vec{R}_2 = 0 \quad (2.5)$$

όπου \vec{R}_1 και \vec{R}_2 σήματα εξόδου του νευρωνικού δικτύου. Για $\vec{x} = [\vec{x}_1 \ \vec{x}_2]$ και $\vec{R} = [\vec{R}_1 \ \vec{R}_2]$, το σύστημα γίνεται:

$$\frac{\partial \vec{x}}{\partial t} = \vec{R} = 0 \quad (2.6)$$

και η μέθοδος Newton πολλαπλών μεταβλητών είναι:

$$J(\vec{x}^{(k)}) \cdot \vec{\delta}^{(k)} = -R(\vec{x}^{(k)}) \quad (2.7)$$

με J την ιακωβιανή ορίζουσα του διανύσματος \vec{R} και $\vec{\delta}$ ένα διάνυσμα που δίνει τη μεταβολή κάθε τιμής του x σε κάθε επανάληψη k . Η παραπάνω σχέση είναι της μορφής $A \cdot x = b$ και επιλύεται, για κάθε επανάληψη, με τη μέθοδο GMRES, με σκοπό την εύρεση του $\vec{\delta}$. Εν συνεχεία, εφαρμόζεται η σχέση:

$$\vec{x}^{(k+1)} = \vec{\delta}^{(k)} + \vec{x}^{(k)} \quad (2.8)$$

Για $k = 0$, το $\vec{x}^{(0)}$ δεν αποτελεί παρά διάνυσμα με τις αρχικές εκτιμήσεις. Η μέθοδος GMRES αποτελεί επαναληπτική μέθοδο, οπότε το υπολογιστικό σχήμα Newton-GMRES αποτελείται από δύο επαναληπτικά σχήματα, ένα για τη μέθοδο GMRES που ονομάζεται εσωτερικό και ένα για τη μέθοδο Newton που ονομάζεται εξωτερικό. Η μέθοδος GMRES συνίσταται στον υπολογισμό ενός διανύσματος r που ονομάζεται υπόλοιπο (*residual*) και για αρχική εκτίμηση $\delta^{(0)}$ της λύσης της εξίσωσης $J \cdot \delta = -R$, είναι:

$$r^{(0)} = -R - J\delta^{(0)} \quad (2.9)$$

Σημειώνεται ότι οι εκθέτες αναφέρονται πλέον στον αριθμό της εσωτερικής επανάληψης. Έπειτα, με την εσωτερική επαναληπτική μέθοδο, επιχειρείται μείωση του r κάτω από ένα συγκεκριμένο όριο πραγματοποιώντας υπολογισμούς στον υπόχωρο Krylov (Krylov subspace). Για δ την υπό αναζήτηση λύση, υπάρχει z τέτοιο ώστε, $\delta = \delta^{(0)} + z$ και η σχέση $J \cdot \delta = -R$ μετατρέπεται σε: $Jz = r^{(0)}$. Η μέθοδος GMRES λοιπόν, χρησιμοποιώντας το γινόμενο της ιακωβιανής ορίζουσας με ένα διάνυσμα, δεν απαιτεί την πλήρη γνώση του πίνακα, ο οποίος, σε ένα σύστημα δεκάδων εξισώσεων (για κάθε σημείο στο οποίο χωρίζεται το διάστημα z , ισχύουν οι εξισώσεις 2.4 (α) και 2.4(β)) είναι δύσκολο να υπολογιστεί ενώ και η αποθήκευσή του από τον αλγόριθμο αυξάνει το χρόνο επίλυσης.

Οι δοκιμές που πραγματοποιήθηκαν αφορούσαν εύρεση της μόνιμης κατάστασης, επιβάλλοντας τις συνοριακές συνθήκες της ενότητας 3.1 για τρεις διαφορετικές αρχικές συνθήκες, IC1, IC4 και IC6 (βλέπε πίνακα 2.9) και αριθμό $Da = 0,12$ (δηλαδή τιμή της παραμέτρου τέτοια, ώστε να υπάρχουν δύο δυνατά ζεύγη ευσταθών τιμών για τη μόνιμη κατάσταση).

4. Εφαρμογή μεθοδολογίας σε περιβάλλον Matlab

4.I. Εφαρμογή διεργασιών Gauss

Ο κώδικας Matlab που χρησιμοποιήθηκε στα πλαίσια της παρούσης εργασίας για την εφαρμογή διεργασιών Gauss μπορεί να βρεθεί στο Παράρτημα (B.1). Με το συγκεκριμένο κώδικα επιλέγονται από τα διανύσματα δεδομένων u , u_t , u_x και u_{xx} (τα οποία έχουν τη μορφή στήλης) τρεις χιλιάδες τετράδες τιμών όπως προαναφέρθηκε. Προηγείται αφαίρεση όλων των τιμών NaN που έχουν προκύψει κατά την εισαγωγή των δεδομένων στο Matlab από το Excel, κυρίως από τίτλους και περιγραφές που χώριζαν τα δεδομένα. Έπειτα, σχηματίζεται ο πίνακας `mat_1` των μεταβλητών u , u_x , u_{xx} που θεωρείται ότι επηρεάζουν τη μεταβλητή u_t . Πυρήνα του κώδικα αποτελεί η συνάρτηση `fitrgp` παράγει ένα μοντέλο παλινδρόμησης με διεργασία Gauss χρησιμοποιώντας τα δεδομένα του πίνακα `mat_1` με συνάρτηση kernel την ARD. Οι παράμετροι της συνάρτησης kernel του μοντέλου (πλην της τελευταίας) δίνουν την επίδραση κάθε μεταβλητής επί της μεταβλητής u_t . Μεγαλύτερες τιμές αντιστοιχούν σε μικρότερη συνεισφορά.

4.II. Εκπαίδευση νευρωνικών δικτύων

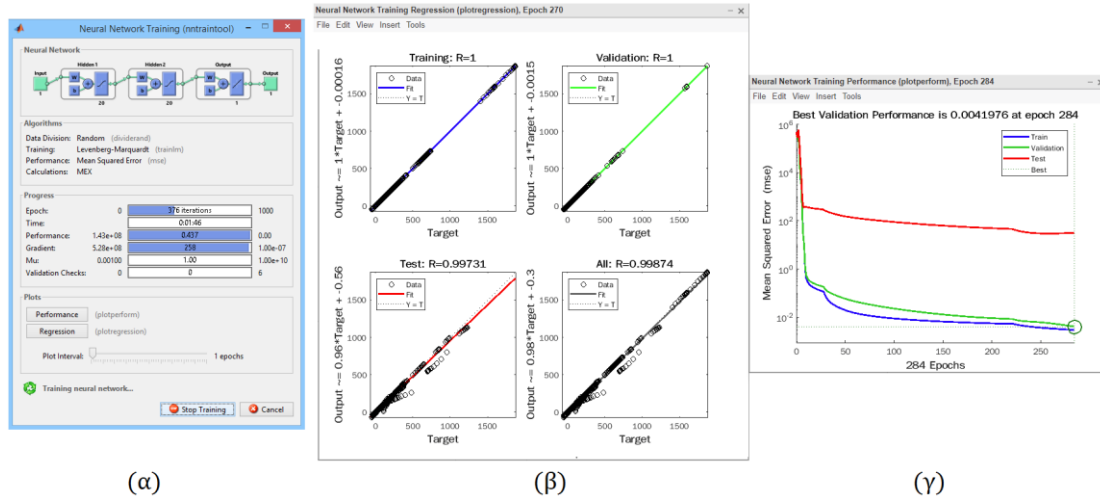
Το Παράρτημα (B.2) περιλαμβάνει τον κώδικα που αξιοποιήθηκε για την εκπαίδευση όλων των νευρωνικών δικτύων που παρήχθησαν στα πλαίσια της συγκεκριμένης εργασίας. Ο κώδικας δημιουργήθηκε αυτόματα χρησιμοποιώντας την εφαρμογή Neural Net Fitting της εργαλειοθήκης Neural Network Toolbox (έκδοση 11.1) του Matlab και κατόπιν έγιναν σε αυτόν οι απαραίτητες τροποποιήσεις. Στον κώδικα ορίζονται:

1. Τα δεδομένα εισόδου και εξόδου που θα χρησιμοποιηθούν για την εκπαίδευση. Τα δεδομένα αυτά περιέχονται στους πίνακες `mat_1` και `mat_2` αντίστοιχα τους οποίους έχει προηγουμένως ορίσει ο χρήστης.
2. Η συνάρτηση εκπαίδευσης.
Ως συνάρτηση εκπαίδευσης χρησιμοποιείται η συνάρτηση `trainlm`, η οποία εφαρμόζει τον αλγόριθμο εκπαίδευσης Levenberg-Marquardt backpropagation. Πυρήνα του αλγορίθμου αποτελεί η μέθοδος gradient descent η οποία περιγράφεται στο κεφάλαιο I (ενότητα 2.2). Ο συγκεκριμένος αλγόριθμος έχει το πλεονέκτημα ότι αποτελεί το γρηγορότερο της εργαλειοθήκης Neural Network Toolbox, για προβλήματα προσέγγισης συνάρτησης, με μέχρι μερικές εκατοντάδες συντελεστών βαρύτητας. Στα πλαίσια της παρούσης εργασίας ο μέγιστος αριθμός συντελεστών βαρύτητας N_w που χρησιμοποιήθηκαν σε ένα δίκτυο ήταν $N_w = 1000$ και ο αριθμός αυτός πληροί το κριτήριο των μερικών εκατοντάδων.
3. Η αρχιτεκτονική του νευρωνικού δικτύου, δηλαδή ο αριθμός των εσωτερικών επιπέδων (hidden layers) και ο αριθμός των νευρώνων κάθε επιπέδου.
4. Η συνάρτηση προεπεξεργασίας (pre-processing) των σημάτων εισόδου και η συνάρτηση μετεπεξεργασίας (post-processing) των σημάτων εξόδου.
Η προεπεξεργασία αποτελεί διαδικασία κατά την οποία τα σήματα εισόδου κανονικοποιούνται ώστε να βρίσκονται εντός συγκεκριμένου εύρους. Αυτό διευκολύνει την εκπαίδευση του νευρωνικού δικτύου καθώς αποτρέπει την εμφάνιση πολύ μικρών κλίσεων (gradients) που θα απαιτούσαν την πραγματοποίηση περισσότερων επαναλήψεων από τον αλγόριθμο εκπαίδευσης. Πραγματοποιείται εν προκειμένω με τη συνάρτηση `mapminmax` η οποία κανονικοποιεί τα δεδομένα στο διάστημα $[-1 \ 1]$. Όμοια, με τη μετεπεξεργασία πραγματοποιείται η αντίστροφη διαδικασία ώστε τα σήματα εξόδου να έχουν κλίμακα σύμφωνη με τα δεδομένα εισόδου.
5. Ο τρόπος και το μέγεθος της διαίρεση των δεδομένων εισόδου σε ομάδες.
Τα δεδομένα που χρησιμοποιούνται για εκπαίδευση χωρίζονται σε τρεις ομάδες (set). Στην πρώτη ομάδα (training set) περιέχονται τα δεδομένα με τα οποία πραγματοποιείται η εκπαίδευση του νευρωνικού δικτύου. Η δεύτερη ομάδα (validation set) χρησιμοποιείται για να τερματίζει την εκπαίδευση του νευρωνικού δικτύου όταν το σφάλμα επαλήθευσης δεν μειώνεται περαιτέρω. Η τρίτη ομάδα (test set) μπορεί να περιέχει δεδομένα που για την εκπαίδευση θεωρούνται άγνωστα. Δεν επηρεάζουν την εκπαιδευτική διαδικασία, αλλά χρησιμοποιούνται για τον έλεγχο της ικανότητα γενίκευσης του δικτύου.
6. Η συνάρτηση σφάλματος.
Εν προκειμένω χρησιμοποιείται η συνάρτηση mean squared error (mse), όπως περιγράφεται στη θεωρία.
7. Οι συναρτήσεις ενεργοποίησης κάθε εσωτερικού επιπέδου.

Σε κάθε νευρωνικό δίκτυο που εκπαιδεύτηκε σε αυτή την εργασία χρησιμοποιήθηκε η σιγμοειδής συνάρτηση \tanh , δηλαδή η συνάρτηση υπερβολικής εφαπτομένης:

$$\varphi(v_k) = \tanh(v_k) = \frac{2}{1 + e^{-2v_k}} - 1 \quad (2.10)$$

Η εκπαίδευση πραγματοποιείται με τη συνάρτηση `train` χρησιμοποιώντας τις παραπάνω επιλογές. Η συνάρτηση ανοίγει επίσης διεπιφάνεια επαφής (GUI) η οποία παρέχει πληροφορίες (όπως ο αριθμός επανάληψης (epoch), η τιμή του σφάλματος (performance) καθώς και κάποιες παράμετροι του αλγορίθμου Levenberg-Marquardt (gradient, mu)) και έχει τη δυνατότητα εμφάνισης επιλεγμένων διαγραμμάτων (η επιλογή πραγματοποιείται στον κώδικα με χρήση των συναρτήσεων `plotperform` και `plotregression`). Η διεπιφάνεια εκπαίδευσης και τα διαγράμματα παρουσιάζονται στην παρακάτω εικόνα:



Εικόνα 2.1. (α) Διεπιφάνεια εκπαίδευσης νευρωνικού δικτύου. (β) Διάγραμμα παλινδρόμησης χωρισμένο σε τέσσερα τμήματα (Training, Validation, Test και αθροιστικό). Το διάγραμμα Test είναι το πλέον χρήσιμο. (γ) Διάγραμμα τετραγωνικού σφάλματος για τις τρεις ομάδες δεδομένων.

4.III. Επίλυση διαφορικής εξίσωσης με χρήση μοντέλου νευρωνικού δικτύου και ode solver

Στο Παράρτημα (B.3) παρατίθενται οι κώδικες που χρησιμοποιήθηκαν για την επίλυση διαφορικής εξίσωσης η οποία εκφραζόταν μέσω μοντέλου νευρωνικού δικτύου. Το μοντέλο υπολόγιζε τις τιμές $\partial y/\partial t$ της εξίσωσης (2.3) λαμβάνοντας ως σήματα εισόδου τιμές των μεταβλητών y και y_{xx} . Από τους τέσσερις κώδικες (script) του παραρτήματος (B.3), ο πρώτος χρησιμοποιείται ανεξαρτήτως συνοριακών συνθηκών και σε αυτόν ορίζονται οι αρχικές συνθήκες (y_0), το διάστημα της ανεξάρτητης μεταβλητής x στο οποίο πραγματοποιείται επίλυση μέσω ode solver καθώς και το χρονικό διάστημα επίλυσης. Ως ode solver χρησιμοποιήθηκε η συνάρτηση `ode15s` του Matlab, η οποία δέχεται ως ορίσματα τη συνάρτηση υπολογισμού του χρονικού διαφορικού εκφρασμένη ως function handle ($f = @(t,y)$), το χρονικό διάστημα επίλυσης και τις αρχικές συνθήκες. Η συνάρτηση παράγει ένα πίνακα $[t,y]$, με την πρώτη στήλη να περιέχει τις χρονικές στιγμές στις οποίες απαιτείται εξαγωγή δεδομένων και τόσες επιπλέον στήλες, όσες τα σημεία του διαστήματος x (δηλαδή a), οι οποίες περιέχουν τις υπολογισμένες τιμές του y .

Όσον αφορά τους υπόλοιπους τρεις κώδικες του παραρτήματος (B.3), χρησιμοποιούνται αναλόγως του είδους των συνοριακών συνθηκών που ο χρήστης επιθυμεί να επιβάλει. Οι συνοριακές συνθήκες υπολογίζονται στην περίπτωση Neumann και Robin χρησιμοποιώντας ένα βοηθητικό σημείο μέσω πεπερασμένων διαφορών. Έστω δηλαδή συνοριακή συνθήκη Neumann $\partial y/\partial x = c$, για $x = 0$. Η εξίσωση (2.3) στο σημείο $x = 0$ (δείκτης 1) με βοηθητικό το σημείο y_0 γίνεται:

$$\frac{\partial y_1}{\partial t} = \frac{y_2 - 2 \cdot y_1 + y_0}{dx^2} + y_1^2 \quad (2.11)$$

Αλλά από τη συνοριακή συνθήκη ισχύει:

$$\frac{\partial y_1}{\partial x} = \frac{(y_2 - y_0)}{dx} = c \Rightarrow (y_2 - y_0) = c \cdot dx \Rightarrow y_0 = y_2 - c \cdot dx \quad (2.12)$$

Έτσι, η (2.11), γίνεται:

$$\frac{\partial y_1}{\partial t} = \frac{y_2 - 2 \cdot y_1 + (y_2 - c \cdot dx)}{dx^2} + y_1^2 = \frac{2 \cdot (y_2 - y_1) - c \cdot dx}{dx^2} + y_1^2 \quad (2.13)$$

Όμοια, για $\partial y/\partial x = c$ στο σημείο $x = a$, η παράγωγος $\partial y/\partial t$ με χρήση του βοηθητικού σημείου y_{a+1} γίνεται:

$$\frac{\partial y_a}{\partial t} = \frac{2 \cdot (y_{a-1} - y_a) + c \cdot dx}{dx^2} + y_a^2 \quad (2.14)$$

Οι σχέσεις για την περίπτωση χρήσης συνοριακών συνθηκών Robin προκύπτουν με την ίδια μέθοδο και είναι:

$$\frac{\partial y_1}{\partial t} = \frac{2 \cdot (y_2 - y_1 - c_1 \cdot dx \cdot y_1 + c_1 \cdot c_2 \cdot dx)}{dx^2} + y_1^2 \quad (2.15)$$

για $x = 0$ με συνοριακή συνθήκη $\partial y/\partial x = c_1 (y_1 + c_2)$ και:

$$\frac{\partial y_a}{\partial t} = \frac{2 \cdot (y_{a-1} - y_a + c_3 \cdot dx \cdot y_a - c_3 \cdot c_4 \cdot dx)}{dx^2} + y_a^2 \quad (2.16)$$

για $x = a$ με συνοριακή συνθήκη $\partial y/\partial x = c_3 (y_a + c_4)$.

4.IV. Εύρεση μεταβατικής κατάστασης με χρήση μοντέλου νευρωνικού δικτύου και ode solver σε πρόβλημα αλωτού αντιδραστήρα

Οι κώδικες για τον υπολογισμό της μεταβατικής κατάστασης στο πρόβλημα αλωτού αντιδραστήρα λειτουργούν με τον ίδιο τρόπο όπως παραπάνω. Οι υπολογισμοί των μερικών διαφορικών ως προς χρόνο στα σημεία $z = 0$ και $z = 1$, δίνονται πλέον από τις σχέσεις:

Για $z = 0$:

$$\frac{\partial x_{1(1)}}{\partial x} = \frac{1}{Pe} \cdot \frac{2 \cdot (x_{1(2)} - x_{1(1)} - Pe \cdot dz \cdot x_{1(1)})}{dz^2} - Pe \cdot x_{1(1)} + Da \cdot (1 - x_{1(1)}) \cdot \exp\left(\frac{x_{2(1)}}{1 + \frac{x_{2(1)}}{\gamma}}\right) \quad (2.17)$$

$$\frac{\partial x_{2(1)}}{\partial x} = \frac{1}{Pe} \cdot \frac{2 \cdot (x_{2(2)} - x_{2(1)} - Pe \cdot dz \cdot x_{2(1)})}{dz^2} - Pe \cdot x_{2(1)} - \beta \cdot x_{2(1)} + B \cdot Da \cdot (1 - x_{1(1)}) \cdot \exp\left(\frac{x_{2(1)}}{1 + \frac{x_{2(1)}}{\gamma}}\right) \quad (2.18)$$

Για $z = 1$:

$$\frac{\partial x_{1(a)}}{\partial x} = \frac{1}{Pe} \cdot \frac{2 \cdot (x_{1(a-1)} - x_{1(a)})}{dz^2} + Da \cdot (1 - x_{1(a)}) \cdot \exp\left(\frac{x_{2(a)}}{1 + \frac{x_{2(a)}}{\gamma}}\right) \quad (2.19)$$

$$\frac{\partial x_{2(a)}}{\partial x} = \frac{1}{Pe} \cdot \frac{2 \cdot (x_{2(a-1)} - x_{2(a)})}{dz^2} - \beta \cdot x_{2(a)} + B \cdot Da \cdot (1 - x_{1(a)}) \cdot \exp\left(\frac{x_{2(a)}}{1 + \frac{x_{2(a)}}{\gamma}}\right) \quad (2.20)$$

4.V. Εύρεση μόνιμης κατάστασης αυλωτού αντιδραστήρα με χρήση Newton-GMRES

Η εύρεση της μόνιμης κατάστασης στο σύστημα που εξετάζεται πραγματοποιείται στο Matlab χρησιμοποιώντας τους δύο κώδικες που επισυνάπτονται στο παράρτημα (Δ). Ο δεύτερος αποτελεί τη συνάρτηση υπολογισμού των μερικών διαφορικών ως προς το χρόνο μέσω του μοντέλου νευρωνικού δικτύου, όπως και στο πρόβλημα μεταβατικής κατάστασης με μικρές αλλαγές για να προκύπτουν διανύσματα με κατάλληλη μορφή. Ο πρώτος κώδικας καλεί τη συνάρτηση gmres η οποία δέχεται ως ορίσματα μία αρχική πρόβλεψη της λύσης x_0 , τη συνάρτηση υπολογισμού του residual (-Res) καθώς και μία συνάρτηση atn (εκφρασμένη ως function handle) η οποία υπολογίζει το γινόμενο του ιακωβιανού πίνακα με διάνυσμα x . Η συνάρτηση αυτή δέχεται επίσης τρεις παραμέτρους που καθορίζουν την ακρίβεια (εκφρασμένη ως μείωση του αρχικού residual, χρησιμοποιήθηκε η τιμή 10^{-2}), το μέγιστο αριθμό επαναλήψεων (εν προκειμένω 20) και την ύπαρξη και το είδος μεθόδου ορθογωνιοποίησης. Οι κώδικες των συναρτήσεων gmres, atn καθώς και givarr και dirder που αξιοποιούνται εσωτερικά της συνάρτησης gmres ελήφθησαν από: Kelley, C. T., Solving Nonlinear Equations with Newton's Method, 1987. Η συνάρτηση gmres παράγει για κάθε εξωτερική επανάληψη ένα διάνυσμα dx , τη μεταβολή δηλαδή του δ ($J\delta = -R$) και ο κώδικας σταματά όταν έχει επιτευχθεί η επιθυμητή ακρίβεια (υπάρχουν επίσης περιπτώσεις απόκλισης (στο άπειρο) ή μη σύγκλισης (μη μείωση του σφάλματος κάτω από ένα όριο λόγω ταλάντωσης) στις οποίες ο χρήστης πρέπει να σταματήσει την εκτέλεση κώδικα).

ΚΕΦΑΛΑΙΟ ΤΡΙΤΟ

Παρουσίαση και σχολιασμός αποτελεσμάτων

1. Προσέγγιση συνάρτησης με επιλογή παραμέτρων και επίλυση διαφορικής εξίσωσης

1.1 Αποτελέσματα εφαρμογής διεργασιών Gauss

Σε αυτή την ενότητα παρουσιάζονται τα αποτελέσματα που προέκυψαν από εφαρμογή διεργασιών Gauss σε δεδομένα από επίλυση της εξίσωσης (2.3), μέσω πεπερασμένων στοιχείων, για τέσσερις αρχικές συνθήκες, GP_1 (κανονική κατανομή), SIN_1 (ημιτονοειδής συνάρτηση), PULSE_1 (ορθογώνιος παλμός με λείανση των γωνιών) και TRIG_1 (τριγωνικός παλμός με λείανση των γωνιών). Σε κάθε μία από τις αρχικές συνθήκες (οι εξισώσεις των οποίων αναγράφονται στον πίνακα 2.1) αντιστοιχούν 560 σημεία. Η εξίσωση (2.3), το δεξί σκέλος της οποίας είναι συνάρτηση της δεύτερης μερικής παραγώγου ως προς τη χωρική μεταβλητή x ($\partial^2 y / \partial x^2$) ή αλλιώς y_{xx} και της ανεξάρτητης μεταβλητής (y^2) θεωρείται μη γνωστή. Επιχειρείται η εύρεση των μεταβλητών που επηρεάζουν σημαντικά το δεξί σκέλος εξετάζοντας τρεις μεταβλητές y , y_x και y_{xx} σε κάθε περίπτωση 560 σημεία. Τα αποτελέσματα συνοψίζονται στον ακόλουθο πίνακα:

Πίνακας 3.1					
Είδος αρχικής συνθήκης	Τιμή παραμέτρου (ARD) ανά μεταβλητή		Είδος αρχικής συνθήκης	Τιμή παραμέτρου (ARD) ανά μεταβλητή	
	GP_1	y		$1,57 \cdot 10^1$	SIN_1
y_x		$2,18 \cdot 10^6$	y_x	$3,28 \cdot 10^6$	
y_{xx}		$1,18 \cdot 10^2$	y_{xx}	$7,81 \cdot 10^1$	
PULSE_1	y	$4,71 \cdot 10^1$	TRIG_1	y	$3,87 \cdot 10^1$
	y_x	$8,16 \cdot 10^{10}$		y_x	$3,27 \cdot 10^{12}$
	y_{xx}	$7,95 \cdot 10^1$		y_{xx}	$7,79 \cdot 10^1$

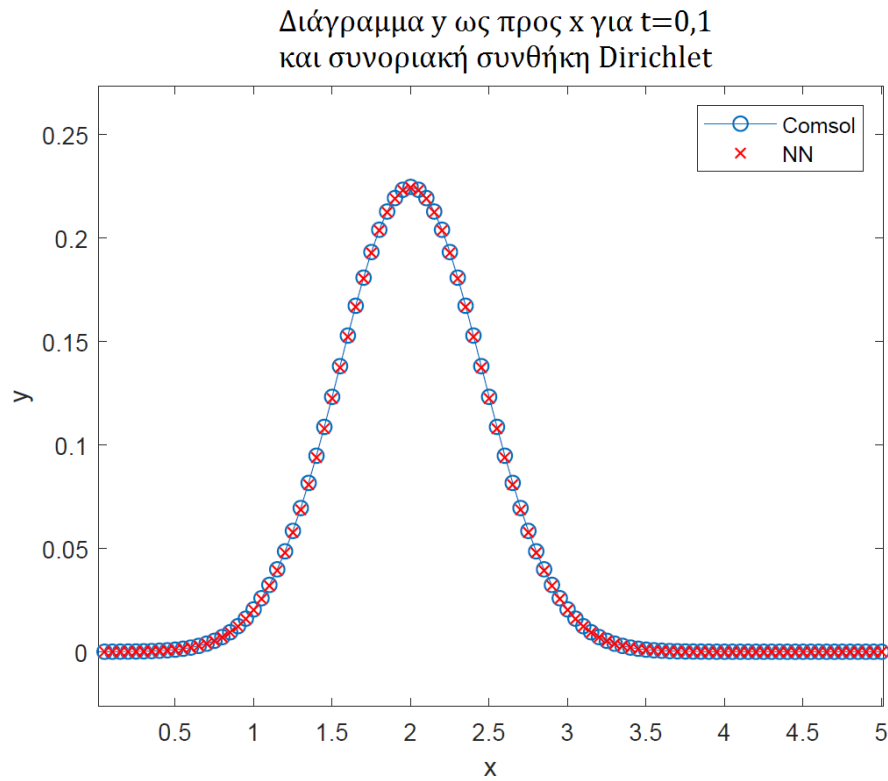
Παρατηρώντας τον πίνακα 3.1, διαπιστώνεται πως οι τιμές των συντελεστών βαρύτητας ARD, για την παράμετρο y_x είναι τουλάχιστον τέσσερις τάξεις μεγέθους χαμηλότερες (στην περίπτωση GP_1) από αυτές των παραμέτρων y και y_{xx} . Έτσι, συνάγεται ότι η πρώτη παράγωγος ως προς τη χωρική μεταβλητή, η οποία σε ένα πραγματικό πρόβλημα θα αποτελούσε όρο συναγωγής, δεν ασκεί στατιστικά σημαντική επίδραση στη χρονική παράγωγο. Αυτό πρακτικά σημαίνει ότι μπορεί να αγνοηθεί κατά την εκπαίδευση μοντέλου νευρωνικού δικτύου το οποίο θα χρησιμοποιείται για τον υπολογισμό μόνιμης και μεταβατικής κατάστασης.

1.2 Αποτελέσματα προσέγγισης δεξιού σκέλους διαφορικής εξίσωσης και εφαρμογής ode solver

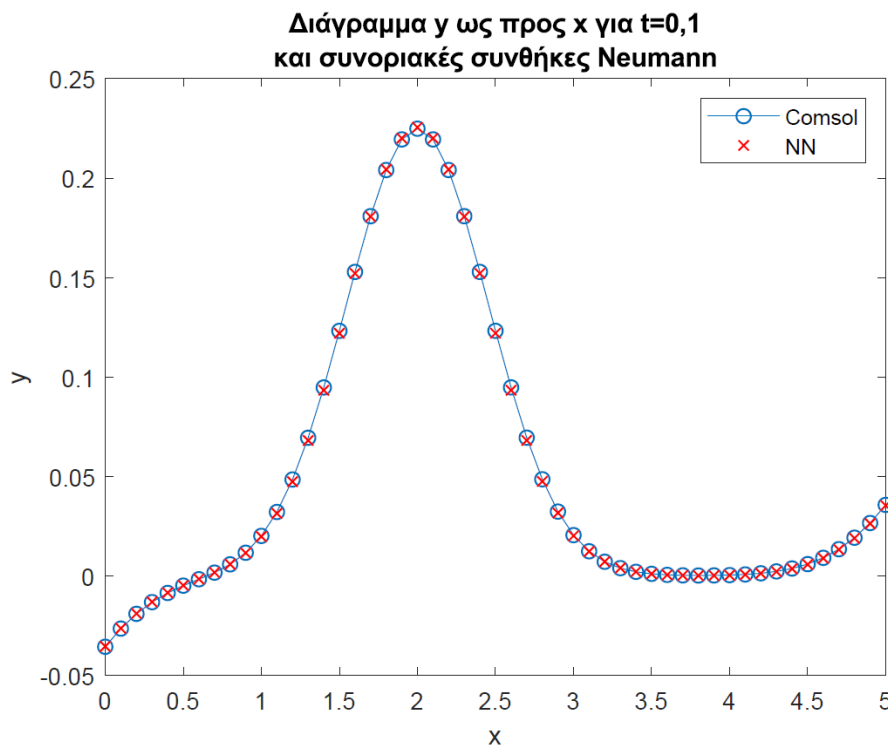
Στην ενότητα αυτή γίνεται παρουσίαση των αποτελεσμάτων που προέκυψαν από εφαρμογή επιλύτη διαφορικών εξισώσεων (ode solver) προβλέποντας τη χρονική παράγωγο της εξίσωσης (2.3) μέσω του μοντέλου νευρωνικού δικτύου NET7000K. Το μοντέλο αυτό είχε τα χαρακτηριστικά του πίνακα 2.3. Γίνεται σύγκριση μεταξύ των αποτελεσμάτων που προέκυψαν από το μοντέλο νευρωνικού δικτύου και από την εφαρμογή μεθόδου πεπερασμένων στοιχείων μέσω COMSOL για τις παρακάτω περιπτώσεις (βλέπε πίνακα 2.4):

- Δοκιμή Π1, αρχική συνθήκη MGAUSS (κανονική κατανομή, βλέπε Κεφ.ΙΙ Ενότητα 2, σελ.15) και συνοριακές συνθήκες Dirichlet.
- Δοκιμή Π2, αρχική συνθήκη MGAUSS και συνοριακές συνθήκες Neumann.
- Δοκιμή Π3, αρχική συνθήκη MGAUSS και συνοριακές συνθήκες Robin.
- Δοκιμή Π4, αρχική συνθήκη MSIGM (σιγμοειδής συνάρτηση, βλέπε Κεφ.ΙΙ Ενότητα 2, σελ.15) και συνοριακές συνθήκες Dirichlet.

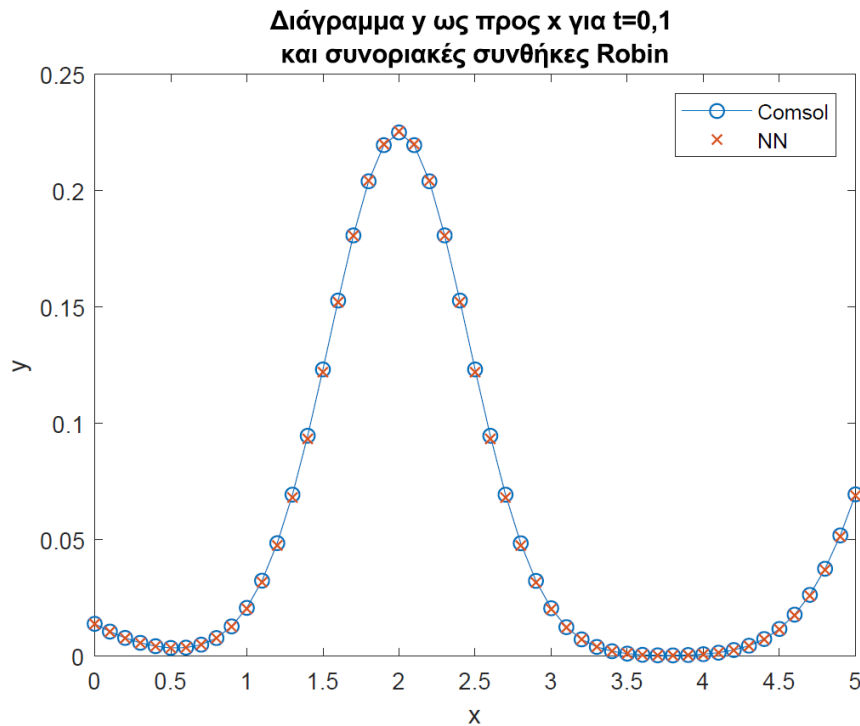
Με σύγκριση των αποτελεσμάτων που προέκυψαν από τις παραπάνω δοκιμές, επιχειρείται να δειχθεί ότι το εκπαιδευμένο μοντέλο μπορεί να χρησιμοποιηθεί ανεξάρτητα του είδους των αρχικών και συνοριακών συνθηκών που επιβάλλονται από το χρήστη, προβλέποντας τη δυναμική των λύσεων του συστήματος. Τα αποτελέσματα που προέκυψαν από τις δοκιμές Π1, Π2 και Π3, για χρόνο $t = 0,1$ φαίνονται στα ακόλουθα διαγράμματα:



Διάγραμμα 3.1. Σύγκριση μεταξύ δεδομένων από μοντέλο NET7000K και Comsol για συνοριακές συνθήκες Dirichlet σε χρόνο $t = 0,1$.



Διάγραμμα 3.2. Σύγκριση μεταξύ δεδομένων από μοντέλο NET7000K και Comsol για συνοριακές συνθήκες Neumann σε χρόνο $t = 0,1$.



Διάγραμμα 3.3. Σύγκριση μεταξύ δεδομένων από μοντέλο NET7000K και Comsol για συνοριακές συνθήκες Robin σε χρόνο $t = 0,1$.

Στο διάγραμμα 3.1 περιέχεται η προβλεπόμενη λύση της εξίσωσης (2.3) τη χρονική στιγμή $t = 0,1$ με κανονική κατανομή ως αρχική συνθήκη ενώ τα διαγράμματα 3.2 και 3.3 αφορούν την ίδια αρχική συνθήκη αλλά συνοριακές συνθήκες Neumann και Robin αντίστοιχα. Σε κάθε περίπτωση το εκπαιδευμένο μοντέλο δεν φαίνεται να εμφανίζει αποκλίσεις που σχετίζονται με επιβολή συνθηκών 2^{ου} και 3^{ου} είδους. Το μέσο σφάλμα για κάθε συνοριακή συνθήκη αναγράφεται στον παρακάτω πίνακα.

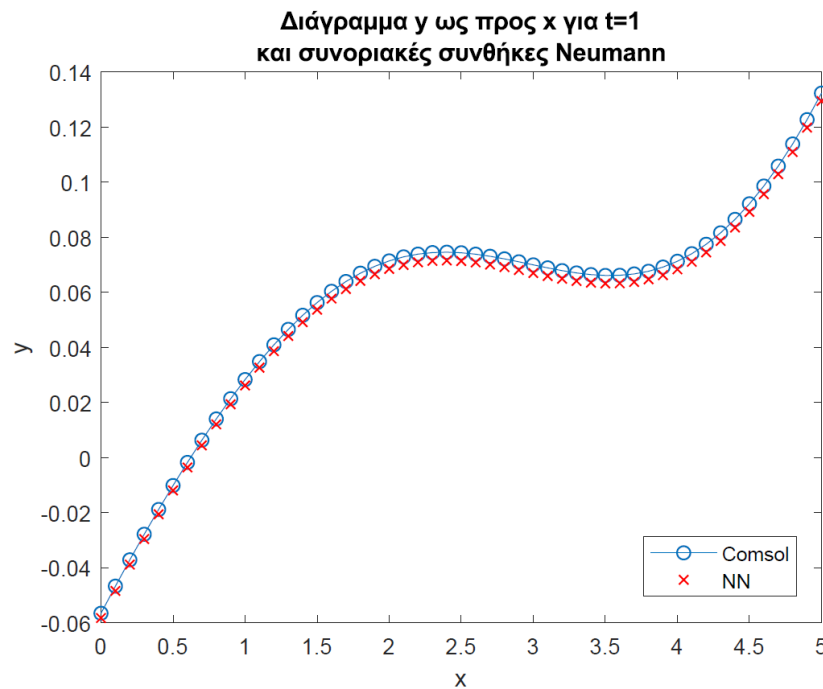
Πίνακας 3.2	Μέσο σφάλμα προσέγγισης λύσης εξίσωσης (2.3) για τρία διαφορετικά είδη συνοριακών συνθηκών
Αριθμός διαγράμματος	Σφάλμα (εκφρασμένο ως mse)
3.1	$1,43 \cdot 10^{-7}$
3.2	$3,43 \cdot 10^{-7}$
3.3	$3,08 \cdot 10^{-7}$

Το μέγιστο σφάλμα των παραπάνω προσεγγίσεων είναι, για κάθε σημείο, μικρότερο από 5 % (εξάιρεση αποτελούν τέσσερα σημεία, με πραγματικές τιμές του y πολύ κοντά στο μηδέν). Από τα παρουσιαζόμενα αποτελέσματα, το μέσο σφάλμα για κάθε συνοριακή συνθήκη είναι της τάξης του 10^{-7} . Γίνεται αντιληπτό, πως το είδος των συνοριακών συνθηκών που χρησιμοποιούνται για την επίλυση της εξίσωσης, δεν επηρεάζει τη δυναμική του μοντέλου.

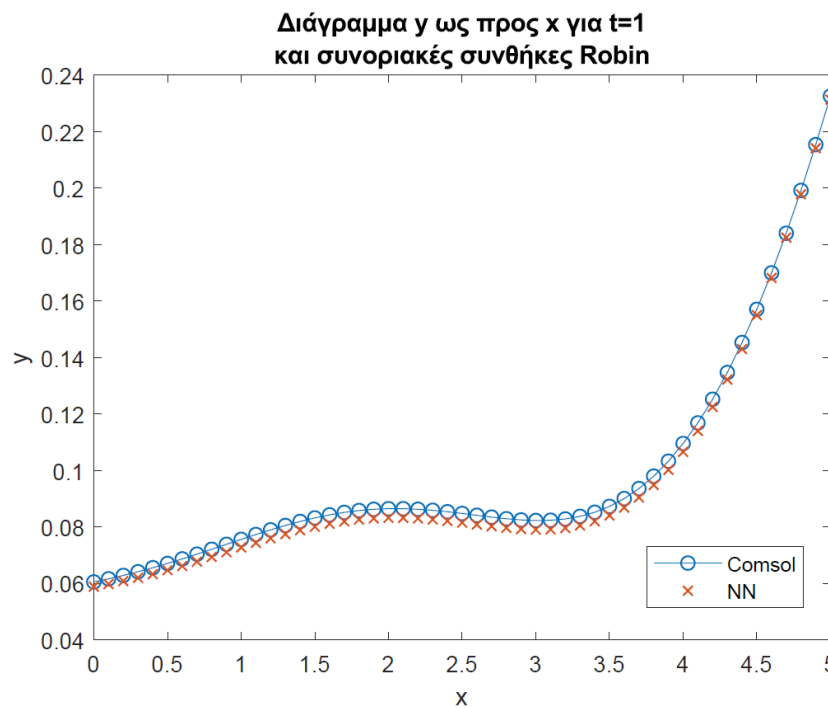
Το παραπάνω συμπέρασμα ισχύει για κάθε χρονική στιγμή. Έτσι, τα αποτελέσματα των δοκιμών Π2 και Π3, για $t = 1$, δηλαδή το μέγιστο χρόνο για τον οποίο παρήχθησαν δεδομένα εκπαίδευσης, τα σφάλματα ήταν:

Πίνακας 3.3	Μέσο σφάλμα προσέγγισης λύσης εξίσωσης (2.3) για συνοριακές συνθήκες Neumann και Robin, σε χρόνο $t = 1$:
Αριθμός διαγράμματος	Σφάλμα (εκφρασμένο ως mse)
3.4	$7,20 \cdot 10^{-6}$
3.5	$7,99 \cdot 10^{-6}$

Τα διαγράμματα 3.4 και 3.5 που αναφέρονται στον πίνακα 3.3, παρατίθενται στην επόμενη σελίδα.



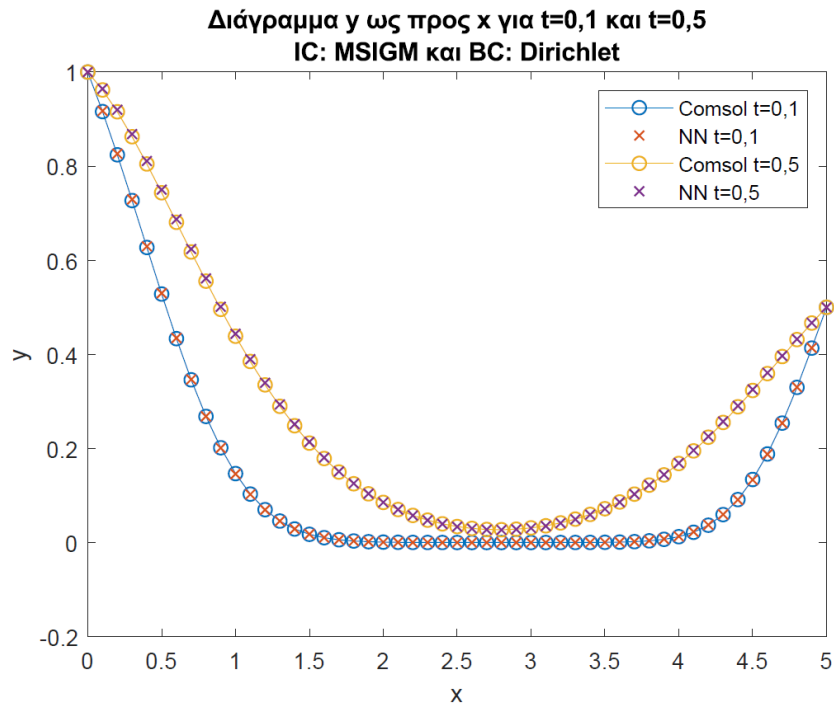
Διάγραμμα 3.4. Σύγκριση μεταξύ δεδομένων από μοντέλο NET7000K και Comsol για συνοριακές συνθήκες Neumann σε χρόνο $t = 1$.



Διάγραμμα 3.5. Σύγκριση μεταξύ δεδομένων από μοντέλο NET7000K και Comsol για συνοριακές συνθήκες Robin σε χρόνο $t = 1$.

Από τα διαγράμματα 3.4 και 3.5, στα οποία γίνεται σύγκριση μεταξύ της πραγματικής λύσης (μέσω Comsol) και της παραχθείσης μέσω του εκπαιδευμένου μοντέλου, αποδεικνύεται ότι δεν υπάρχει επίδραση των συνοριακών συνθηκών στις αποκλίσεις που δίνει το μοντέλο, κατά μήκος του χρονικού διαστήματος επίλυσης. Το μέγιστο σφάλμα εξακολουθεί και στις δύο περιπτώσεις να είναι μικρότερου του 5 %.

Με τα αποτελέσματα της δοκιμής Π4, που παρουσιάζονται στο ακόλουθο διάγραμμα, όπου χρησιμοποιήθηκε ως αρχική συνθήκη η σιγμοειδής συνάρτηση MSIGM, αποδεικνύεται ότι η χαρτογραφημένη σχέση μεταξύ δεξιού και αριστερού σκέλους της διαφορικής εξίσωσης είναι επίσης ανεξάρτητη των αρχικών συνθηκών που επιλέγονται από το χρήστη.



Διάγραμμα 3.6. Σύγκριση μεταξύ δεδομένων από μοντέλο NET7000K και Comsol για συνοριακές συνθήκες Dirichlet σε χρόνους $t = 0,1$ και $t = 0,5$.

Στον παρακάτω πίνακα συνοψίζεται το μέσο σφάλμα για κάθε μία από τις δύο χρονικές στιγμές.

Πίνακας 3.4	Προσέγγιση εξίσωσης (2.3) με οριακές συνθήκες Dirichlet: $y/x=0 = 1$ και $y/x=a = 0,5$ και αρχική συνθήκη MSIGM	
	Χρόνος t	Σφάλμα (εκφρασμένο ως mse)
3.6	0,1	$6,09 \cdot 10^{-7}$
3.6	0,5	$8,28 \cdot 10^{-6}$

Το μέγιστο σφάλμα για χρόνο $t = 0,1$ ήταν κάτω από 3 % ενώ για $t = 0,5$ ήταν χαμηλότερο του 2 %, το δε μέσο σφάλμα, σύμφωνα με τον πίνακα 3.5 ήταν για την πρώτη περίπτωση (χρονική στιγμή $t = 0,1$) ίδιας τάξης μεγέθους (10^{-7}) με το αντίστοιχο, για χρήση κανονικής κατανομής (MGAUSS) ως αρχικής συνθήκης (διάγραμμα 3.1). Έτσι, προκύπτει το συμπέρασμα ότι η μεθοδολογία που αναπτύσσεται στην παρούσα εργασία προσφέρει την ικανότητα πρόβλεψης της λύσης μίας διαφορικής εξίσωσης, ανεξάρτητα από το είδος των αρχικών συνθηκών.

Από την ενότητα αυτή εξάγεται το συμπέρασμα ότι είναι εφικτό να κατασκευαστεί μοντέλο νευρωνικού δικτύου, χρησιμοποιώντας τις παραμέτρους εκείνες που αποδεικνύεται από την εφαρμογή διεργασιών Gauss ότι επηρεάζουν σημαντικά το αποτέλεσμα, το οποίο προσεγγίζοντας το δεξί σκέλος μίας άγνωστης διαφορικής εξίσωσης, μπορεί να προβλέπει την εξέλιξη λύσεων στο χρόνο. Οι λύσεις αυτές, οι οποίες προκύπτουν χρησιμοποιώντας κλασσικές μεθόδους αριθμητικής ανάλυσης δεν φαίνεται να επηρεάζονται από τις επιλεγθείσες αρχικές συνθήκες. Ομοίως, από τα παρουσιαζόμενα αποτελέσματα γίνεται αντιληπτό πως το είδος των συνοριακών συνθηκών που επιβάλλεται για την επίλυση της εξίσωσης, δεν επηρεάζει την απόκλιση του μοντέλου.

2. Υπολογισμός μεταβατικής κατάστασης για διαφορετικές αρχικές συνθήκες, σε πρόβλημα αυλωτού αντιδραστήρα και σύγκριση με δεδομένα από Comsol.

2.1 Σύγκριση αποτελεσμάτων μεταξύ μοντέλου NN και Comsol για τιμή παραμέτρου $Da = 0,12$.

Σε αυτή την ενότητα παρουσιάζονται τα αποτελέσματα πρόβλεψης λύσεων του συστήματος μερικών διαφορικών εξισώσεων (2.4), το οποίο (αντικαθιστώντας τις σταθερές βάσει του πίνακα 2.5) είναι:

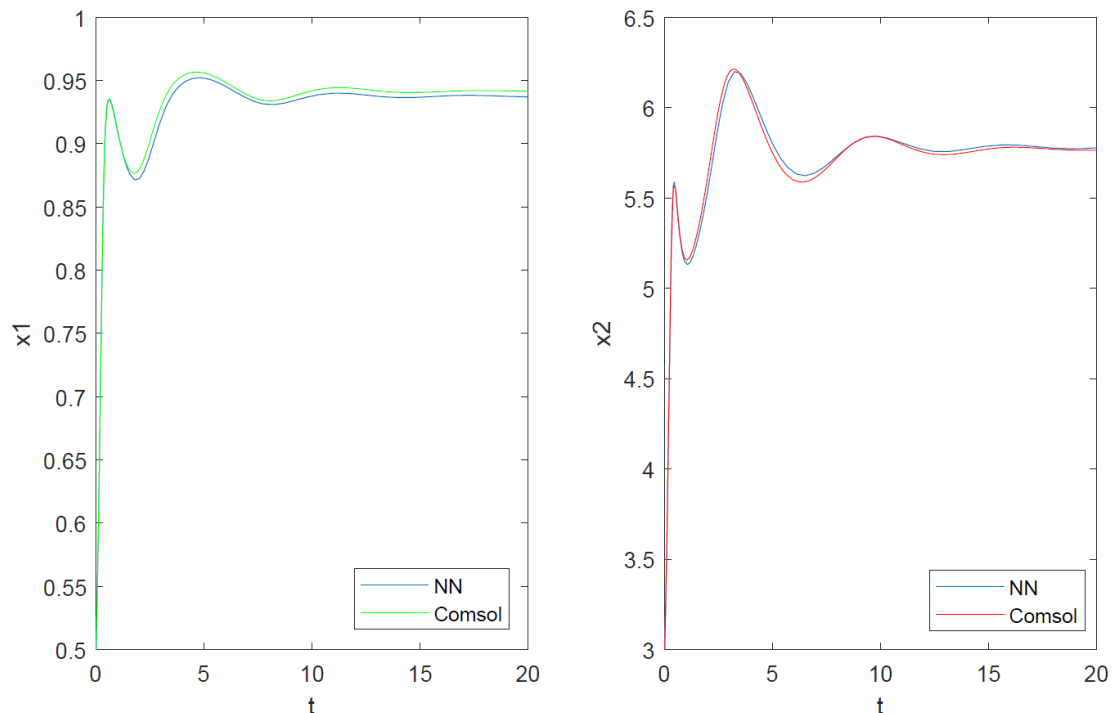
$$\frac{\partial x_1}{\partial t} = 0,2 \frac{\partial^2 x_1}{\partial z^2} - \frac{\partial x_1}{\partial z} + Da(1 - x_1) \exp\left(\frac{x_2}{1 + 0,05 \cdot x_2}\right)$$

$$\frac{\partial x_2}{\partial t} = 0,2 \frac{\partial^2 x_2}{\partial z^2} - \frac{\partial x_2}{\partial z} - 1,5 \cdot x_2 + 12 \cdot Da(1 - x_1) \exp\left(\frac{x_2}{1 + 0,05 \cdot x_2}\right)$$

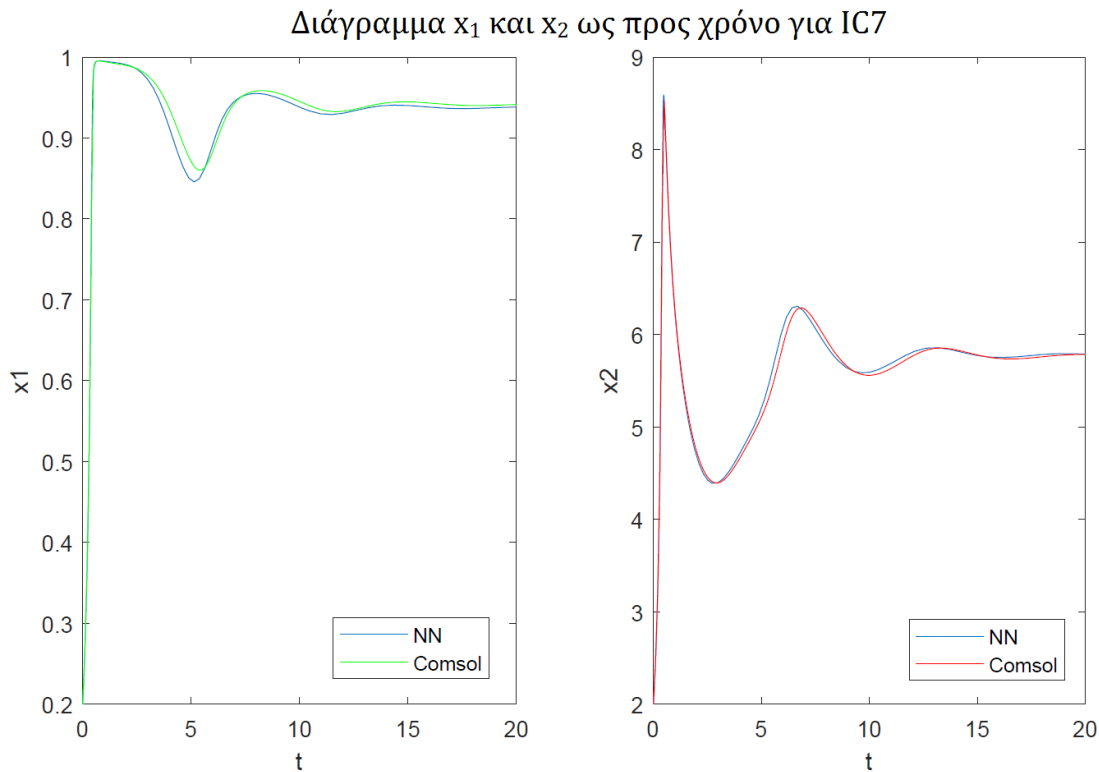
Οι προσεγγίσεις έγιναν για το χρονικό διάστημα $[0, 20]$, χρησιμοποιώντας τιμή παραμέτρου $Da = 0,12$. Η τιμή αυτή επιλέχθηκε ώστε, να εξασφαλίζεται ότι το σύστημα εμφανίζει πολλαπλότητα λύσεων και συνεπώς να διαπιστωθεί αν το εκπαιδευμένο μοντέλο είναι σε θέση να συγκλίνει προς την πραγματική μόνιμη κατάσταση. Το μοντέλο, το οποίο χρησιμοποιήθηκε για την παραγωγή των αποτελεσμάτων αυτής της ενότητας αυτής, ήταν το NET3020_1_160, με τα χαρακτηριστικά του πίνακα 2.8. Τα δεδομένα εκπαίδευσης τα οποία χρησιμοποιήθηκαν, αφορούσαν έντεκα τιμές του αριθμού Da , από 0,08 έως 0,130 με βήμα 0,005. Ο μέγιστος χρόνος για τον οποίο παρήχθησαν δεδομένα ήταν $t = 0,5$ δηλαδή σαράντα φορές μικρότερος από το μέγιστο χρόνο για τον οποίο υπολογίστηκε η μεταβατική κατάσταση. Έτσι γίνεται έλεγχος της ικανότητας γενίκευσης του μοντέλου και πρόβλεψης σε μεγάλους χρόνους.

Στα ακόλουθα διαγράμματα φαίνεται η μεταβατική κατάσταση, στο σημείο $z = 1$, για αρχική συνθήκη IC4 ($x_1 = 0,5$ και $x_2 = 3$) και IC7 ($x_1 = 0,2$ και $x_2 = 2$).

Διάγραμμα x_1 και x_2 ως προς χρόνο για IC4



Διάγραμμα 3.7. Σύγκριση x_1 , x_2 μεταξύ NET3020_1_160 και Comsol για χρόνους στο διάστημα $[0,20]$, για IC4.



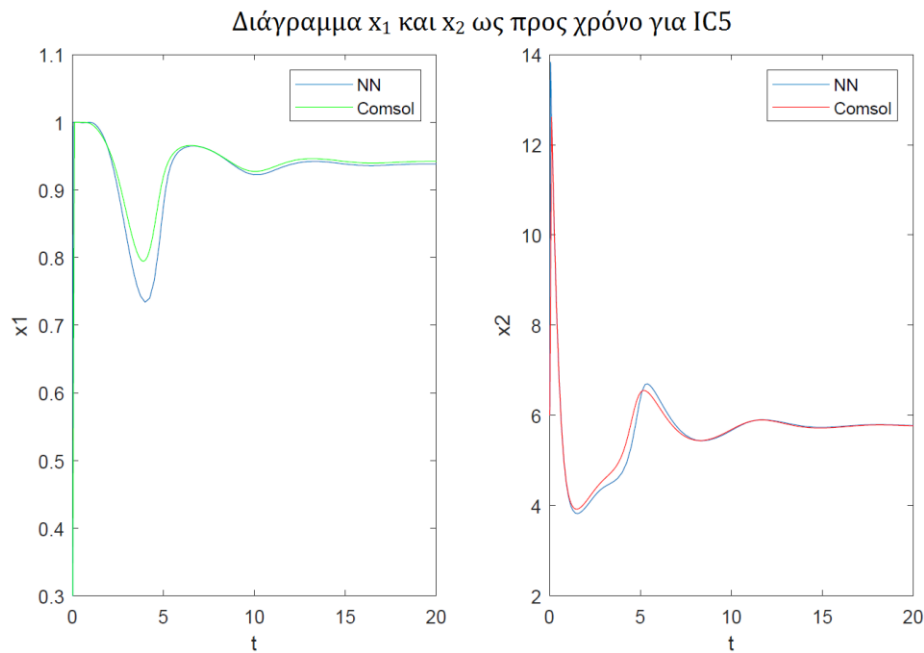
Διάγραμμα 3.8. Σύγκριση x_1 , x_2 μεταξύ NET3020_1_160 και Comsol για χρόνους στο διάστημα $[0,20]$, για IC7.

Από τα διαγράμματα 3.7 και 3.8 αποδεικνύεται πως με το εκπαιδευμένο μοντέλο επιτυγχάνεται με ακρίβεια ο υπολογισμός της δυναμικής του συστήματος για δύο αρχικές συνθήκες, διαφορετικές από αυτές που χρησιμοποιήθηκαν για την εκπαίδευση του μοντέλου και σε μεγάλους χρόνους. Για το διάγραμμα 3.7 η μέγιστη απόκλιση του x_1 είναι 1,25 % και του x_2 1,23 %. Ομοίως, για αρχική IC7 (διάγραμμα 3.8) οι μέγιστες αποκλίσεις των x_1 και x_2 είναι 3,95 % και 5,30 % αντίστοιχα.

Στα πλαίσια της παρούσης εργασίας πραγματοποιήθηκαν επιπλέον υπολογισμοί, χρησιμοποιώντας όλες τις αρχικές συνθήκες που αναφέρονται στον πίνακα 2.9. Στον ακόλουθο πίνακα παρουσιάζονται τα σφάλματα για κάθε αρχική συνθήκη.

Αρχική συνθήκη	Μέσο και μέγιστο σφάλμα ως προς x_1 και x_2 για IC1-IC7 και $Da = 0,12$			
	Σφάλμα			
	mse, για το σύνολο των σημείων (50)		μέγιστο %	
	x_1	x_2	x_1	x_2
IC1	$1,82 \cdot 10^{-5}$	$0,47 \cdot 10^{-2}$	1,55	3,54
IC2	$4,31 \cdot 10^{-4}$	$1,77 \cdot 10^{-2}$	9,35	9,10
IC3	$5,39 \cdot 10^{-4}$	$2,24 \cdot 10^{-2}$	10,40	10,28
IC4	$1,96 \cdot 10^{-5}$	$0,16 \cdot 10^{-2}$	1,25	1,23
IC5	$5,69 \cdot 10^{-4}$	$2,37 \cdot 10^{-2}$	10,80	10,70
IC6	$3,83 \cdot 10^{-6}$	$2,25 \cdot 10^{-2}$	1,63	2,59
IC7	$1,03 \cdot 10^{-4}$	$0,74 \cdot 10^{-2}$	3,95	5,30

Όπως φαίνεται στον πίνακα 3.5, για τρεις από τις αρχικές συνθήκες που δοκιμάστηκαν το μέγιστο σφάλμα είναι περίπου 10 %. Για παράδειγμα για αρχική συνθήκη IC5 ($x_1 = 0,3$ και $x_2 = 6$) και $z = 1$, προέκυψε το ακόλουθο διάγραμμα.



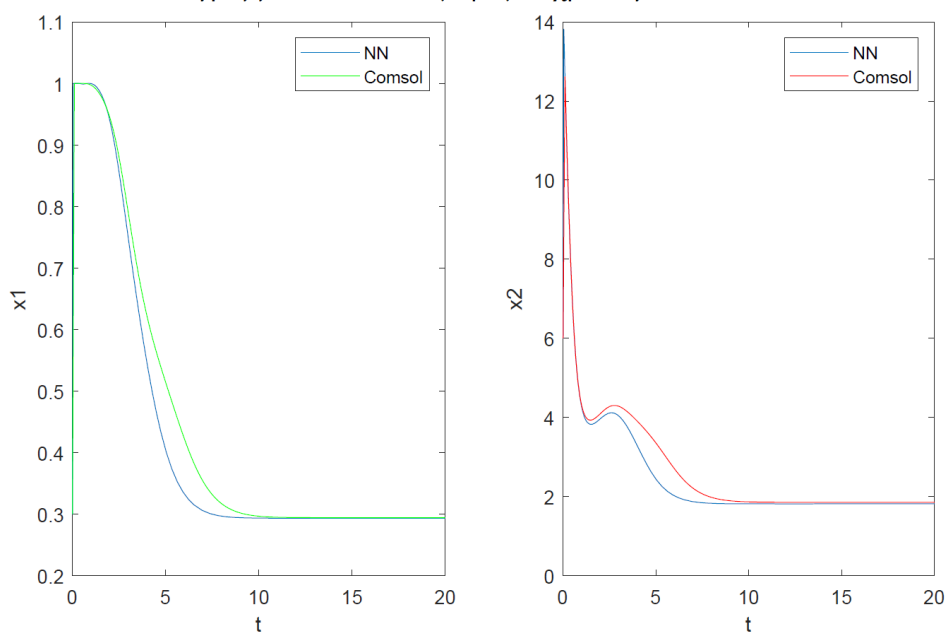
Διάγραμμα 3.9. Σύγκριση x_1 , x_2 μεταξύ NET3020_1_160 και Comsol για χρόνους στο διάστημα $[0,20]$, για IC5.

Τα μέγιστα σφάλματα εντοπίζονται (και για τις περιπτώσεις που δεν παρουσιάζονται εν προκειμένω) κοντά σε χρόνο $t = 5$. Ωστόσο, όπως φαίνεται από το διάγραμμα 3.9, η προβλεπόμενη λύση, γύρω από χρόνο $t = 5$ έχει μορφή παραπλήσια με την πραγματική. Έτσι, αν και το μοντέλο παρουσιάζει αποκλίσεις, αυτές δεν εμποδίζουν την εύρεση της δυναμικής του συστήματος.

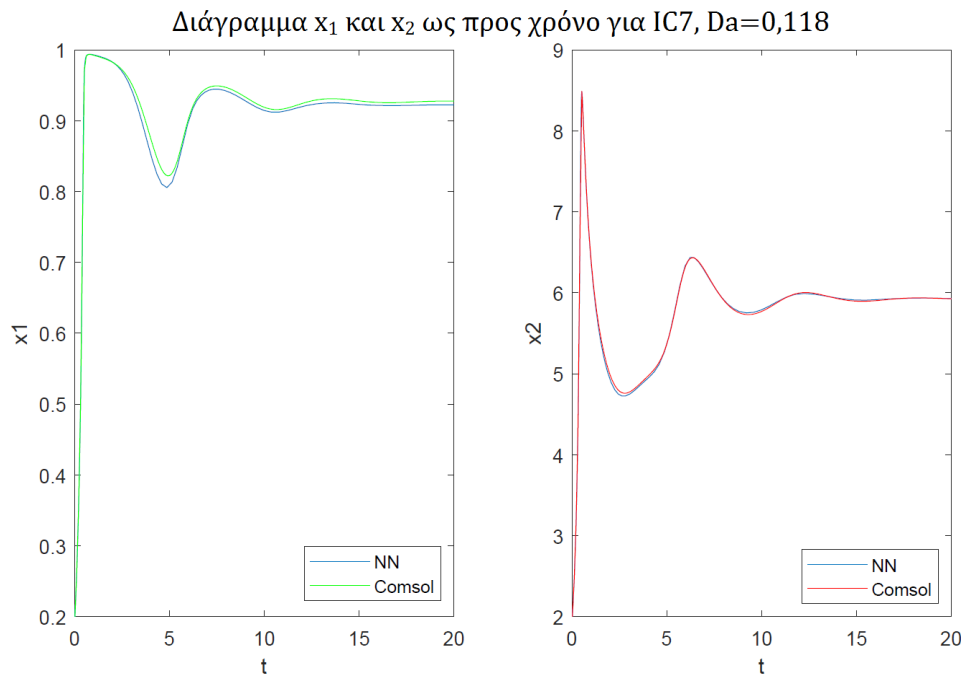
2.2 Σύγκριση αποτελεσμάτων μεταξύ μοντέλου NN και Comsol για τιμή παραμέτρου $Da = 0,118$.

Στην ενότητα αυτή παρουσιάζονται τα αποτελέσματα της ίδιας διαδικασίας, με αυτή που ακολουθήθηκε στην ενότητα 2.1 του παρόντος κεφαλαίου, πλην όμως, για τιμή παραμέτρου $Da = 0,118$. Επειδή χρησιμοποιήθηκαν για την εκπαίδευση του μοντέλου νευρωνικού δικτύου NET3020_1_160, δεδομένα για $Da = 0,115$ και $Da = 120$, γίνεται έλεγχος της ικανότητας του μοντέλου για πρόβλεψη της δυναμικής των λύσεων σε ενδιάμεση τιμή. Για αρχικές συνθήκες IC5 ($x_1 = 0,3$ και $x_2 = 6$) και IC7 ($x_1 = 0,2$ και $x_2 = 2$), τα αποτελέσματα φαίνονται στα παρακάτω διαγράμματα.

Διάγραμμα x_1 και x_2 ως προς το χρόνο για IC5, $Da=0,118$



Διάγραμμα 3.10. Σύγκριση x_1 , x_2 μεταξύ NET3020_1_160 και Comsol για χρόνους στο διάστημα $[0,20]$, για IC5 και $Da = 0,118$.



Διάγραμμα 3.11. Σύγκριση x_1, x_2 μεταξύ NET3020_1_160 και Comsol για χρόνους στο διάστημα $[0,20]$, για IC7 και $Da = 0,118$.

Τα διαγράμματα 3.10 και 3.11 δείχνουν τη σύγκριση της δυναμικής της λύσης, στο σημείο $z = 1$, μεταξύ πρόβλεψης με το εκπαιδευμένο μοντέλο και δεδομένων από Comsol για λύσεις σε δύο διαφορετικούς κλάδους. Η λύση από μοντέλο νευρωνικού δικτύου στην περίπτωση του διαγράμματος 3.10 φαίνεται να παρουσιάζει τις μεγαλύτερες αποκλίσεις, γύρω από το χρόνο $t = 5$. Τα σφάλματα υπολογίστηκαν:

Αρχική συνθήκη	Μέσο και μέγιστο σφάλμα ως προς x_1 και x_2 για IC5 και IC7 και $Da = 0,118$			
	Σφάλμα			
	mse, για το σύνολο των σημείων (50)		μέγιστο %	
	x_1	x_2	x_1	x_2
IC5	$0,20 \cdot 10^{-2}$	$1,19 \cdot 10^{-1}$	24,13	29,67
IC7	$7,08 \cdot 10^{-5}$	$8,50 \cdot 10^{-4}$	3,03	1,10

Από τα δεδομένα του πίνακα 3.6 και ειδικότερα το μέγιστο σφάλμα περίπου 30 % στην περίπτωση της αρχικής συνθήκης IC5, γίνεται αντιληπτό πως η χαμηλή πυκνότητα τιμών παραμέτρου Da , που χρησιμοποιήθηκε για την παραγωγή δεδομένων εκπαίδευσης επηρέασε την ικανότητα του μοντέλου για πρόβλεψη σε τιμές Da χωρίς δεδομένα. Ωστόσο, και στις δύο περιπτώσεις (διαγράμματα 3.10 και 3.11) η προβλεπόμενη λύση φαίνεται να διατηρεί τουλάχιστον σχήμα όμοιο με την πραγματική.

Από τα αποτελέσματα της ενότητας αυτής, αποδεικνύεται πως το εκπαιδευμένο μοντέλο NET3020_1_160 είναι ικανό για ακριβή πρόβλεψη της μεταβατικής κατάστασης του συστήματος διαφορικών εξισώσεων (2.4) έχοντας εκπαιδευθεί με δεδομένα που αντιστοιχούσαν σε χρόνους πολύ μικρότερους από αυτούς που χρησιμοποιήθηκαν για τον έλεγχο του μοντέλου ($t = 0,5$ έναντι $t = 20$ αντίστοιχα).

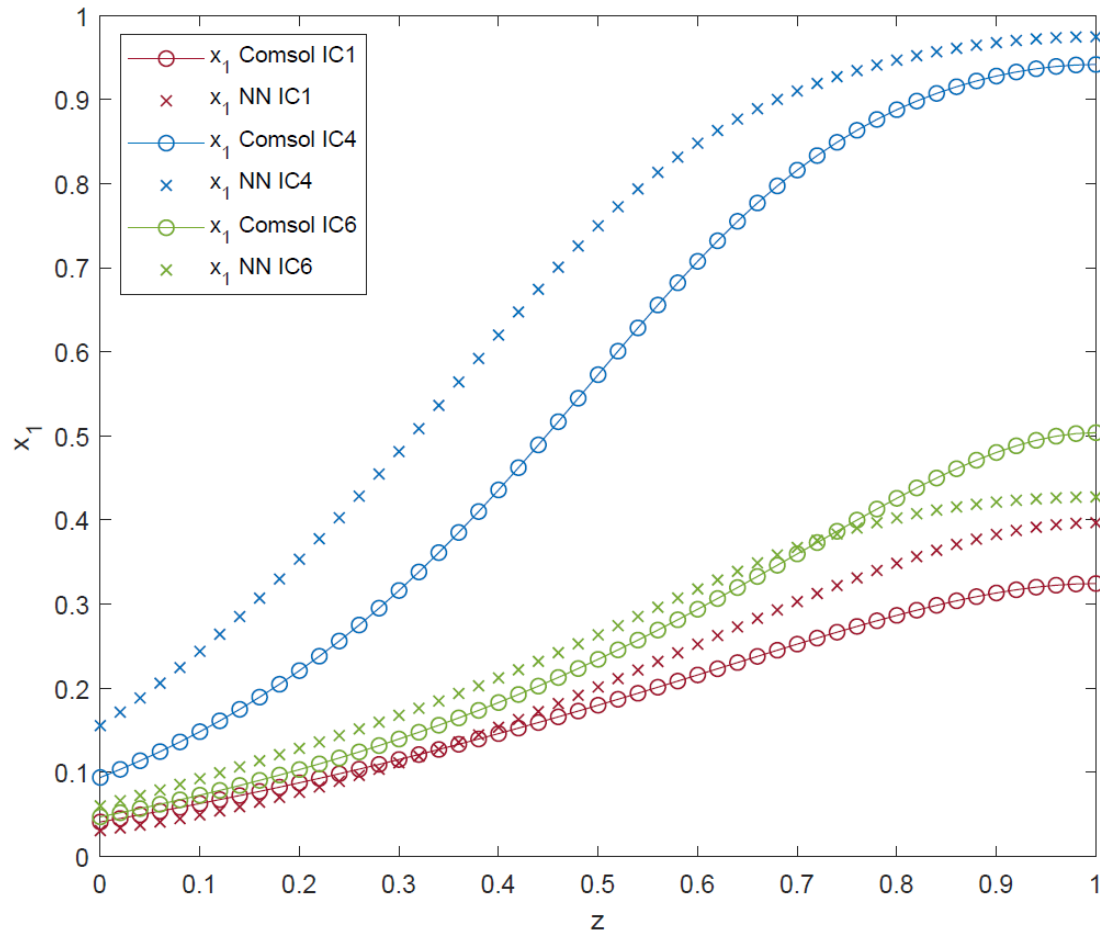
3. Υπολογισμός μόνιμης κατάστασης σε περιοχή πολλαπλότητας λύσεων, σε πρόβλημα αυλωτού αντιδραστήρα και σύγκριση με δεδομένα από Comsol.

Στην παρούσα ενότητα παρατίθενται τα αποτελέσματα που προέκυψαν από την εφαρμογή της μεθοδολογίας εύρεσης μόνιμης κατάστασης (ενότητα 3.2, Κεφάλαιο ΙΙ), συνδυάζοντας τεχνητό νευρωνικό δίκτυο και μέθοδο Newton-GMRES. Η μόνιμη κατάσταση υπολογίστηκε για τρεις διαφορετικές αρχικές συνθήκες, IC1 ($x_1 = 0,1$ και $x_2 = 1$), IC4 ($x_1 = 0,5$ και $x_2 = 3$) και IC6 ($x_1 = 0,7$ και $x_2 = 2$). Το εκπαιδευμένο μοντέλο που χρησιμοποιήθηκε ήταν το NET3510_2_87, για την εκπαίδευση του οποίου είχαν χρησιμοποιηθεί τα μισά περίπου δεδομένα σε σχέση με το μοντέλο που χρησιμοποιήθηκε στην περίπτωση του μεταβατικού προβλήματος (ενότητα 2, παρόντος κεφαλαίου).

Τα δεδομένα εκπαίδευσης αφορούσαν τιμές του αριθμού Da από 0,08 έως 0,105 με βήμα 0,005 δηλαδή έξι τιμές, με τη μόνιμη κατάσταση να ανήκει σε κάθε περίπτωση στον κάτω κλάδο λύσεων (περιοχή Ι, διαγράμματος 2.2). Εξετάζεται αν το μοντέλο είναι σε θέση να προβλέπει με ακρίβεια το σύνολο των τριών μόνιμων καταστάσεων (δύο ευσταθών και μίας ασταθούς) που εμφανίζει το σύστημα εξισώσεων (2.4) για τιμή του αριθμού $Da = 0,12$ (τιμή όπου εμφανίζεται πολλαπλότητα λύσεων, περιοχή ΙΙ, διαγράμματος 2.2).

Στο παρακάτω διάγραμμα φαίνονται οι τιμές που λαμβάνει το διάνυσμα x_1 στη μόνιμη κατάσταση, κατά μήκος του διαστήματος z (χωρική μεταβλητή), για τις τρεις προαναφερθείσες αρχικές συνθήκες.

Διάγραμμα x_1 στη μόνιμη κατάσταση και σύγκριση με δεδομένα από Comsol, IC1, IC4, IC6



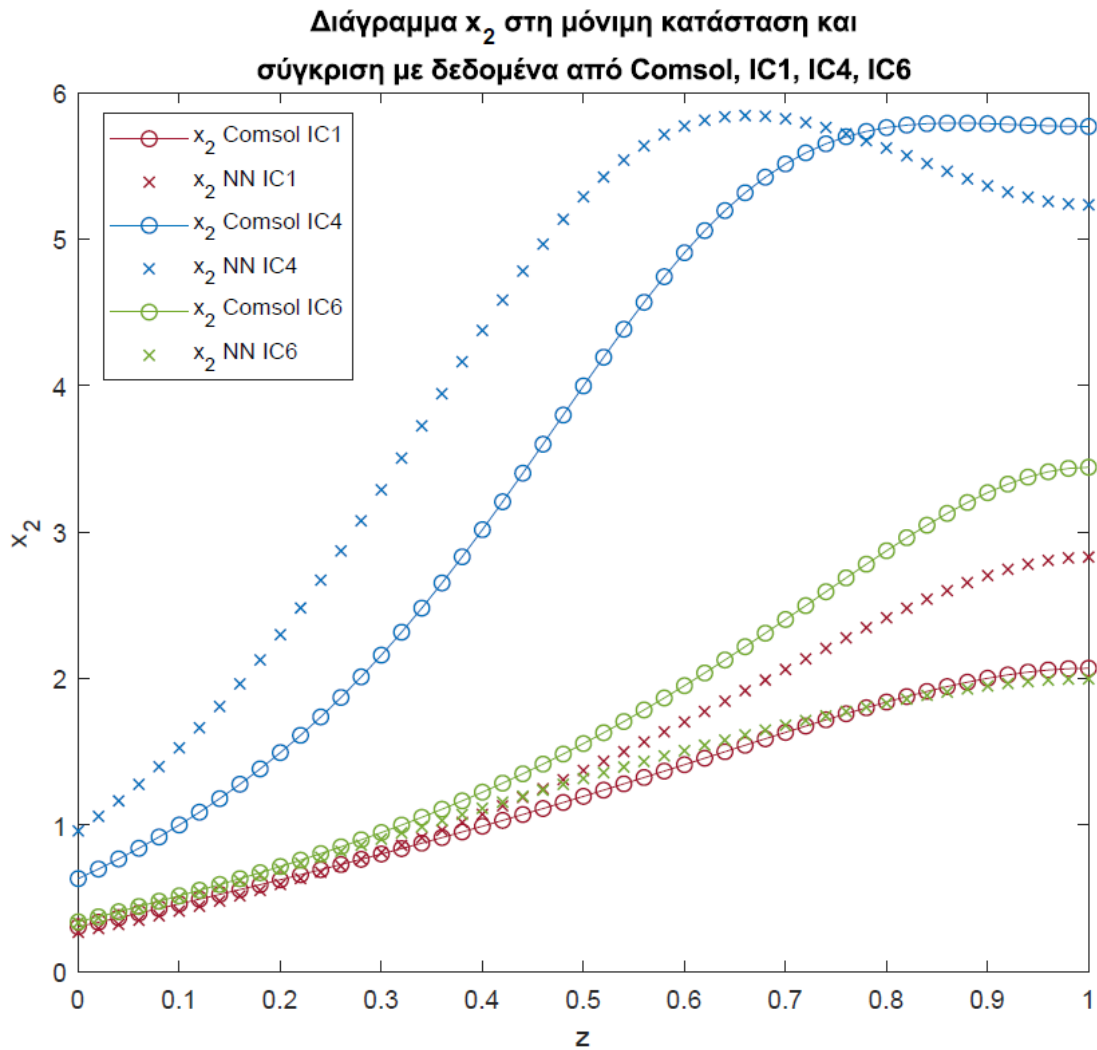
Διάγραμμα 3.13. Σύγκριση λύσεων steady-state για το x_1 στη μόνιμη κατάσταση μεταξύ NET3510_1_87 και Comsol για IC1, IC4, IC6 και αριθμό $Da = 0,12$.

Παρατηρώντας το διάγραμμα 3.13, γίνεται πρωτίστως αντιληπτό ότι το μοντέλο δεν είναι ικανό να προβλέψει με μεγάλη ακρίβεια τη μόνιμη κατάσταση. Τα αποτελέσματα που προέκυψαν από τον υπολογισμό των αποκλίσεων φαίνονται στον παρακάτω πίνακα:

Πίνακας 3.7 Μέγιστο και μέσο σφάλμα επί τοις εκατό, για κάθε αρχική συνθήκη, στον υπολογισμό της μόνιμης κατάστασης της μεταβλητής x_1

Αρχική συνθήκη	Μέγιστο σφάλμα %	Μέσο σφάλμα %
IC1	25,1	16,0
IC4	65,6	32,7
IC6	27,4	15,0

Παρά τις μεγάλες αποκλίσεις, το μοντέλο προβλέπει τρεις διαφορετικές λύσεις. Αυτές, για $z = 1$, είναι 0,40 για IC1, 0,97 για IC4 καθώς και 0,43 για αρχική συνθήκη IC6. Για τη μεταβλητή x_2 , προέκυψε το ακόλουθο διάγραμμα:



Διάγραμμα 3.14. Σύγκριση λύσεων steady-state για το x_2 στη μόνιμη κατάσταση μεταξύ NET3510_1_87 και Comsol για IC1, IC4, IC6 και αριθμό $Da = 0,12$.

Στο διάγραμμα 3.14 φαίνεται η πρόβλεψη, από το μοντέλο νευρωνικού δικτύου, τριών λύσεων, οι οποίες, στο σημείο $z = 1$, είναι 2,83 για αρχική συνθήκη IC1, 5,23 για αρχική συνθήκη IC4 και αντίστοιχα 2,00 για IC6. Η πρόβλεψη του μοντέλου για IC4 προσεγγίζει την πραγματική μόνιμη κατάσταση για IC1, κάτι το οποίο οφείλεται σε σφάλματα του μοντέλου (τόσο στο το διάγραμμα 3.13 όσο και στο 3.14, παρατηρείται ότι το μοντέλο δίνει μεγαλύτερη τιμή από την πραγματική στο $z = 1$, για IC1 και μικρότερη για IC6). Οι αποκλίσεις συνοψίζονται στον επόμενο πίνακα:

Πίνακας 3.8 Μέγιστο και μέσο σφάλμα επί τοις εκατό, για κάθε αρχική συνθήκη, στον υπολογισμό της μόνιμης κατάστασης της μεταβλητής x_2

Αρχική συνθήκη	Μέγιστο σφάλμα %	Μέσο σφάλμα %
IC1	36,7	18,0
IC4	53,9	29,9
IC6	42,0	18,5

Σε μία ρεαλιστική εφαρμογή, με άγνωστη τη διαφορική εξίσωση, τα μεγάλα σφάλματα του μοντέλου στην εύρεση της μόνιμης κατάστασης, ειδικά για IC1 και IC6 θα μπορούσαν να είχαν οδηγήσει σε αβεβαιότητα ως προς τα εάν πρόκειται για μία ή δύο λύσεις. Όμως, υπενθυμίζεται ότι τα δεδομένα που χρησιμοποιήθηκαν προς εκπαίδευση του μοντέλου NET3510_2_87, παρήχθησαν χρησιμοποιώντας πολύ μικρή πυκνότητα τιμών αριθμού Da (μόλις έξι τιμές). Το κυριότερο συμπέρασμα αυτής της ενότητας είναι ότι, ακόμη και με τη χρήση λίγων σχετικά δεδομένων, τα οποία προέρχονταν μόνο από ένα κλάδο λύσεων, η μεθοδολογία της παρούσης εργασίας χρησιμοποιήθηκε επιτυχώς για τη διαπίστωση της ύπαρξης ενός ακόμη κλάδου, για τον οποίο δεν υπήρχε προηγούμενη γνώση.

ΚΕΦΑΛΑΙΟ ΤΕΤΑΡΤΟ

Συμπεράσματα και προτάσεις για μελλοντική μελέτη

1. Προσέγγιση συνάρτησης με επιλογή παραμέτρων και επίλυση διαφορικής εξίσωσης

Όσον αφορά την εφαρμογή διεργασιών Gauss και την προσέγγιση του αριστερού σκέλους διαφορικής εξίσωσης μέσω μοντέλου νευρωνικού δικτύου προκύπτει ότι το εκπαιδευμένο μοντέλο μπορεί να χρησιμοποιηθεί ανεξάρτητα από το είδος των συνοριακών και αρχικών συνθηκών που επιβάλλονται και σε χρόνους όπου δεν υπήρχαν δεδομένα για εκπαίδευση. Η μέγιστες αποκλίσεις σε ορισμένες από τις δοκιμές που πραγματοποιήθηκαν, ήταν μικρότερες του 2 %. Το σημαντικότερο συμπέρασμα που προκύπτει από τη διαδικασία αυτή είναι πως από ένα πείραμα μπορούν να παραχθούν δεδομένα για συγκεκριμένες αρχικές και συνοριακές συνθήκες χωρίς να είναι γνωστή η διαφορική εξίσωση και οι μεταβλητές που επηρεάζουν τη δεξιά σκέλος της. Έπειτα, τα δεδομένα μπορούν να χρησιμοποιηθούν για την εφαρμογή της μεθοδολογίας της παρούσης εργασίας και πρόβλεψη σε πλήθος συνοριακών και αρχικών συνθηκών.

2. Υπολογισμός μεταβατικής κατάστασης σε πρόβλημα αυλωτού αντιδραστήρα.

Το μοντέλο NET3020_1_160 το οποίο χρησιμοποιήθηκε σε αυτή την ενότητα παρουσιάζει ικανότητα γενίκευσης τέτοια, ώστε η ακρίβεια στην εύρεση της μεταβατικής κατάστασης να ξεπερνά σε ορισμένες δοκιμές το 98 %. Το σημαντικότερο συμπέρασμα αυτής της ενότητας είναι πως το εκπαιδευμένο μοντέλο είναι ικανό να χαρτογραφεί τη σχέση μεταξύ δεξιού και αριστερού σκέλους της διαφορικής εξίσωσης και έτσι να επιτυγχάνει ακριβή πρόβλεψη της μεταβατικής κατάστασης για χρόνους σαράντα φορές μεγαλύτερους από το μέγιστο που είχε χρησιμοποιηθεί για εκπαίδευση. Αυτό είναι ουσιαστικής σημασίας σε περιπτώσεις όπου υπάρχουν διαθέσιμα πειραματικά δεδομένα για ένα σύστημα, σε μικρούς χρόνους και ζητείται η πλήρης συμπεριφορά του συστήματος στο χρόνο.

3. Υπολογισμός μόνιμης κατάστασης σε περιοχή πολλαπλότητας λύσεων, σε πρόβλημα αυλωτού αντιδραστήρα.

Με χρήση ενός μοντέλου νευρωνικού δικτύου το οποίο είχε εκπαιδευτεί για μικρούς χρόνους και τιμές παραμέτρου τέτοιες, ώστε να εξασφαλίζεται ότι η μόνιμη κατάσταση του υπό εξέταση συστήματος εξισώσεων βρισκόταν (για κάθε τιμή παραμέτρου) στον ίδιο κλάδο ευσταθών λύσεων, επιχειρήθηκε πρόβλεψη της μόνιμης κατάστασης σε περιοχή πολλαπλότητας λύσεων. Διαπιστώθηκε ότι, αν και με μεγάλα σφάλματα σε σχέση με την προηγούμενη ενότητα, επακόλουθο του μικρότερου αριθμού εκπαιδευτικών δεδομένων που χρησιμοποιήθηκαν σε αυτή την περίπτωση, το εκπαιδευμένο μοντέλο είχε την ικανότητα πρόβλεψης του δεύτερου κλάδου λύσεων, παρά το γεγονός ότι δεν υπήρχε προηγούμενη γνώση για ύπαρξη τέτοιου κλάδου.

Δυστυχώς, με τα μοντέλα που αναπτύχθηκαν στην εν λόγω εργασία δεν κατέστη δυνατή η εύρεση του σημείου στο οποίο η μόνιμη κατάσταση παρουσιάζει ταλάντωση (Hopf bifurcation point). Ένας λόγος για αυτό μπορεί να είναι το μεγάλο βήμα, που χρησιμοποιήθηκε για τις τιμές της παραμέτρου Da ($5 \cdot 10^{-3}$) κατά την παραγωγή δεδομένων. Αναφέρεται (Koronaki, Boudounis & Kevrekidis, 2003) ότι σημείο ταλάντωσης εμφανίζεται για $Da = 0,11003$, απέχοντας μόνο $3 \cdot 10^{-5}$ από την τιμή 0,11, μία εκ των έντεκα για τις οποίες παρήχθησαν δεδομένα.

Προτάσεις για μελλοντική μελέτη

Προτείνεται ως τμήμα μελλοντικής εργασίας η πραγματοποίηση παραμετρικής ανάλυσης, χρησιμοποιώντας εκπαιδευμένα μοντέλα νευρωνικών δικτύων σε συνδυασμό με μεθόδους όπως η pseudo arc-length continuation method (Crysinas, Kavousanakis & Boudounis, 2018), που θα οδηγούσε σε πλήρη χαρτογράφηση του χώρου λύσεων, με διαγράμματα της μορφής 2.2, κάτι το οποίο δεν πραγματοποιήθηκε στα πλαίσια της συγκεκριμένης εργασίας.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Α. Βιβλία

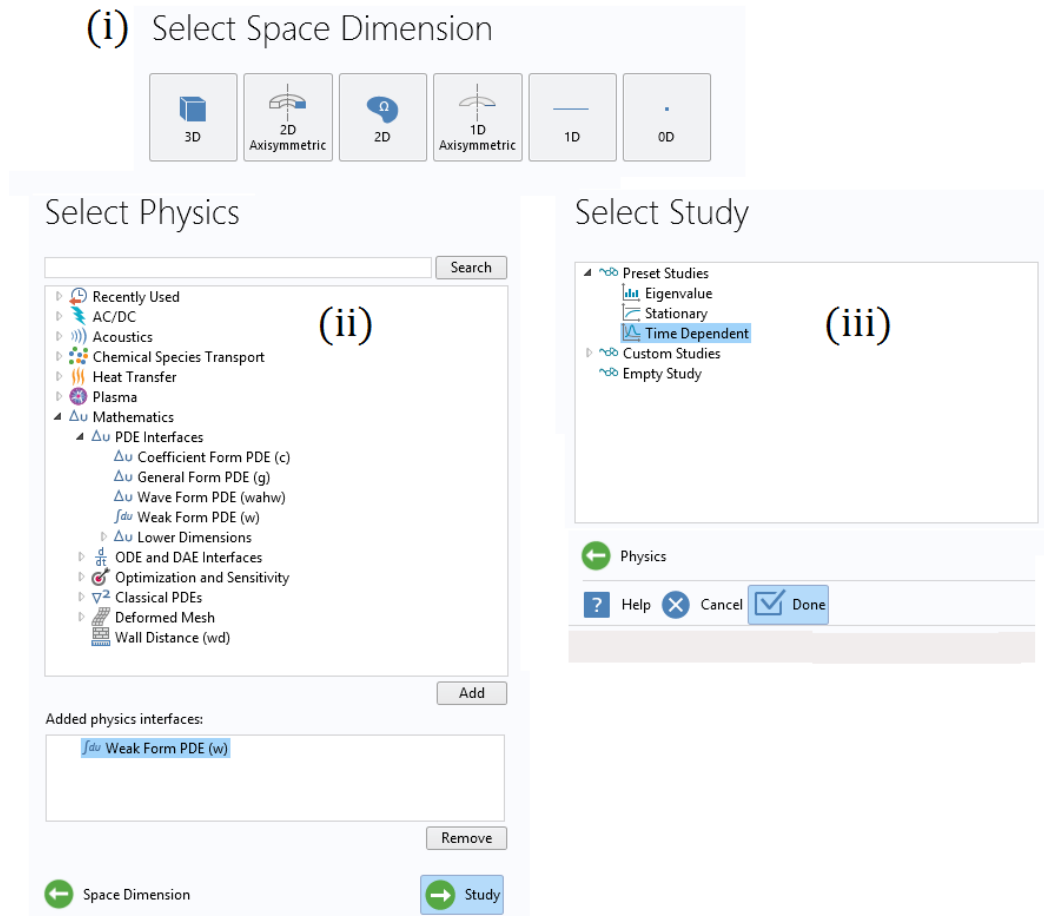
- Boyce W. E. , DiPrima R. C., 1999, Στοιχειώδεις Διαφορικές Εξισώσεις και Προβλήματα Συνοριακών Τιμών, Πανεπιστημιακές εκδόσεις ΕΜΠ, Αθήνα
- Carl Edward Rasmussen & Christopher K. I. Williams, 2006, Gaussian Processes for Machine Learning, The MIT Press, Massachusetts
- Dobkin Daniel M., Zuraw Michael K., 2003, Principles of Chemical Vapor Deposition, Springer-Science+Business Media B.V.
- Haykin, Simon, 1999, Neural Networks: A Comprehensive Foundation, 2nd edition, Pearson Education, Singapore
- Kelley, C. T., 1987, Solving Nonlinear Equations with Newton's Method, Society for Industrial and Applied Mathematics, Philadelphia
- Mitchell, Tom M., 1997, Machine Learning, McGraw-Hill Science/Engineering/Math, USA
- Pierson, Hugh O., 1999, Handbook of chemical vapor deposition, 2nd edition, Noyes Publications / William Andrew Publishing LLC, New York
- Witten Ian H., Eibe Frank, Hall Mark A., Pal Christopher J., 2017, 4th edition, Morgan Kaufmann, Cambridge, USA

Β. Δημοσιεύσεις

- Arbabi, H., Kevrekidis, I.G., 2021, Particles to partial differential equations parsimoniously, Chaos 31, 033137
- Brown, P.N., Saad, Y., 1990, Hybrid Krylov Methods for Nonlinear Systems of Equations, SIAM J. Sci. Stat. Comput., Vol. 11, No. 3, pp. 450-481
- Cybenko, G., 1989, Approximation by Superpositions of a Sigmoidal Function, Math. Control Signals Systems, Vol. 2, pp. 303-314
- Chrysinas, P., Kavousanakis, M.E., Boudouvis, A.G., 2018, Effect of cell heterogeneity on isogenic populations with the synthetic genetic toggle switch network: bifurcation analysis of two-dimensional Cell Population Balance Models, Computers and Chemical Engineering, Vol. 112, pp: 27-36
- Jensen, K.F., Ray, H.W., 1982, The bifurcation behavior of tubular reactors, Chemical Engineering Science, Vol. 37, No. 2, pp. 199-222
- Jordan, M. I. & Mitchell T. M., 2015, Machine learning: Trends, perspectives, and prospects, Science, Vol 349, Issue 6245, pp. 255-260
- Koronaki, E.D., Boudouvis, A.G., Kevrekidis, I.G., 2003, Enabling stability analysis of tubular reactor models using PDE/PDAE integrators, Computers and Chemical Engineering, Vol. 27, pp. 951-964
- LeCun, Y., Bengio, Y. & Hinton, G., 2015, Deep Learning, Nature, Vol. 52, pp. 436-44
- Lee, S., Kooshkbaghi, M., Spiliotis, K., Siettos C. I. and Kevrekidis I. G., 2020, Coarse-scale PDEs from fine-scale observations via machine learning, Chaos 30, 013141
- Raissi, M., Karniadakis, G.E., 2018, Hidden physics models: Machine learning of nonlinear partial differential equations, J. Comput. Phys., Vol. 357, pp. 125-141
- Rosenblatt, F., 1958, The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain, Psychological Review, Vol. 65, No. 6, pp. 386-408

ΠΑΡΑΡΤΗΜΑ Α.

Στην παρακάτω εικόνα συνοψίζεται η δημιουργία μοντέλου μέσω του model wizard του Comsol.

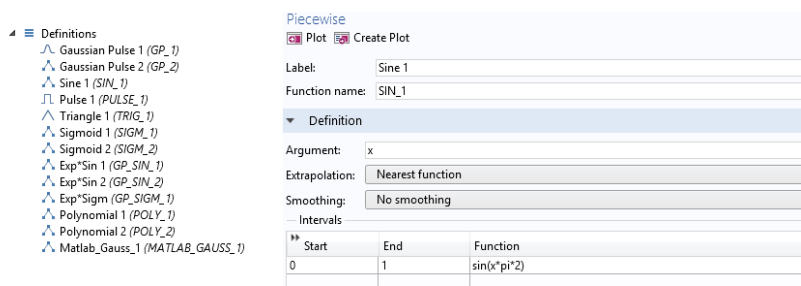


Παραπάνω φαίνονται:

- i) Η επιλογή του αριθμού διαστάσεων που εξαρτάται από το είδος του προβλήματος. Στην παρούσα διπλωματική χρησιμοποιήθηκε η επιλογή ‘1D’.
- ii) Η επιλογή του είδους του προβλήματος. Εν προκειμένω επιλέγεται από την κατηγορία Mathematics, PDE Interfaces και η μορφή στην οποία θα εισαχθεί η διαφορική εξίσωση.
- iii) Η επιλογή του είδους της μελέτης, ‘Time dependent’ για μελέτη στο χρόνο ή ‘Stationary’ για προβλήματα μόνιμης κατάστασης.

Η παραγωγή δεδομένων μέσω του Comsol Multiphysics ακολουθεί τη διαδικασία που περιγράφεται με τα παρακάτω βήματα:

(α) Ορισμός αρχικών συνθηκών.



Στο πεδίο ‘Definitions’ εισάγονται οι εξισώσεις που περιγράφουν τις αρχικές συνθήκες που χρησιμοποιούνται.

(β) Εισαγωγή διαστήματος επίλυσης.

Geometry 1

- Interval 1 (*i1*)
- Form Union (*fin*)

Interval

Number of intervals: One

Left endpoint: 0 m

Right endpoint: 1 m

Πρόκειται για τα άκρα της γραμμής (1D) επί της οποίας θα λυθεί η εξίσωση.

(γ) Σχηματισμός πλέγματος (mesh)

Mesh 1

Mesh Settings

Sequence type: Physics-controlled mesh

Element size: Extremely fine

Όσο πυκνότερο είναι το πλέγμα που χρησιμοποιείται τόσο περισσότερα τα δεδομένα που παράγονται και ενδεχομένως να έχουν μεγαλύτερη ακρίβεια από αραιότερες πυκνότητες (κατόπιν δοκιμών χρησιμοποιήθηκαν πυκνότητες μεγαλύτερες από 'Finer').

(δ) Εισαγωγή εξίσωσης

$\int_{\Omega} \text{Weak Form PDE } (w)$

- Weak Form PDE 1
- Zero Flux 1
- Initial Values 1
- Dirichlet Boundary Condition 1
- Dirichlet Boundary Condition 2

Mesh 1

Equation

Show equation assuming: Study 1, Time Dependent

$$0 = \int_{\Omega} \text{weak } \partial L$$

Weak Expressions

weak `ut*test(u)[m^4]+ux*test(ux)[m^6][1/s]-u^2*test(u)[m^4][1/s]`

Η υπό επίλυση εξίσωση εισάγεται στην ασθενή μορφή (weak formulation) για την παραγωγή δεδομένων της ενότητας 2.A και στην general form για το πρόβλημα αυλωτού αντιδραστήρα της ενότητας 2.B.

(ε) Επιλογή αρχικής συνθήκης

Initial Values 1

Initial Values

Initial value for u:
 u SIGM_1(x[1/m]) 1

Initial time derivative of u:
 $\frac{\partial u}{\partial t}$ 0 1/s

Επιλέγεται μία από τις εισαχθείσες προηγούμενων αρχικές συνθήκες.

(στ) Εισαγωγή συνοριακών συνθηκών

Dirichlet Boundary Condition 1

Dirichlet Boundary Condition 2

Dirichlet Boundary Condition

Prescribed value of u

r 1 1

Graphics Convergence Plot 1

Select Boundaries

Εισάγονται οι συνοριακές συνθήκες επιλέγοντας στο σχήμα και το άκρο στο οποίο ισχύουν.

(ζ) Επιλογή χρονικών στιγμών για εξαγωγή δεδομένων

Study 1

Step 1: Time Dependent

Solver Configurations

Settings

Time Dependent

Compute Update Solution

Label: Time Dependent

Study Settings

Time unit: s

Times: range(0,0.1,0.5) s

Relative tolerance: 0.01

Results While Solving

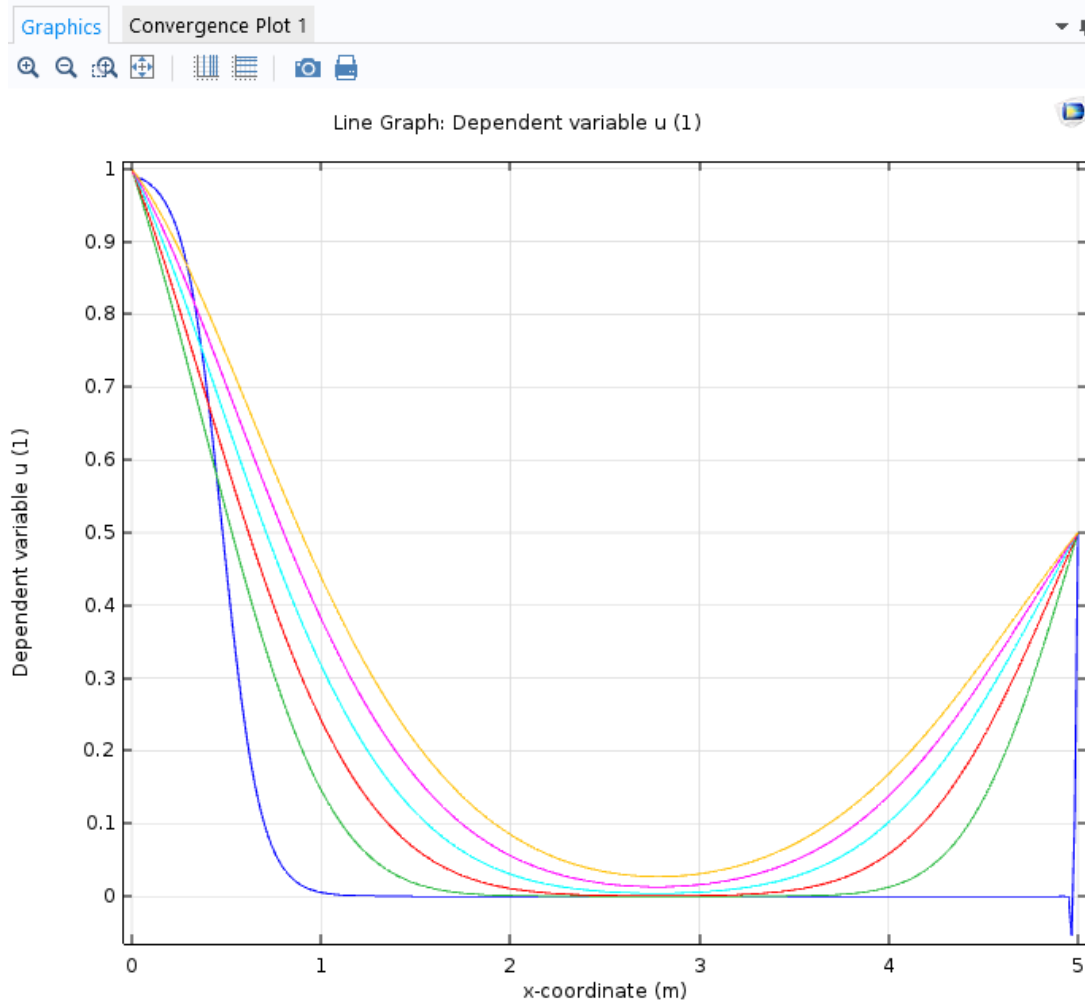
Physics and Variables Selection

Modify physics tree and variables for study step

Physics interface	Solve for	Discretization
Weak Form PDE (w)	<input checked="" type="checkbox"/>	Physics settings

Χρησιμοποιώντας την εντολή range εισάγονται οι χρονικές στιγμές στις οποίες απαιτείται καταγραφή δεδομένων για εξαγωγή. Όπως και στην περίπτωση της ode15s του Matlab, η επίλυση δεν λαμβάνει υπόψιν του χρόνου αυτού παρά μόνο επιστρέφει τιμές χρησιμοποιώντας γραμμική παρεμβολή μεταξύ εσωτερικά υπολογισμένων στιγμών.

(η) Έλεγχος αποτελεσμάτων



Αφού λυθεί η εξίσωση εμφανίζεται διάγραμμα όπως το παραπάνω. Κάθε γραμμή αντιστοιχεί σε μία χρονική στιγμή από τις ζητηθείσες. Από το διάγραμμα ελέγχεται αν η λύση που προέκυψε έχει νόημα.

(θ) Εξαγωγή δεδομένων

Τέλος, από τη διεπιφάνεια που παρουσιάζεται στην επόμενη σελίδα, πραγματοποιείται εξαγωγή δεδομένων σε μορφή αρχείου .txt στο φάκελο που ορίζεται. Τα αρχεία .txt στη συνέχεια εισάγονται σε φύλλα Excel όπου υφίστανται κατάλληλες τροποποιήσεις (π.χ. αφαίρεση τιμών κοντά στις συνοριακές συνθήκες, προσθήκη τίτλων) ώστε να είναι ευδιάκριτα από το χρήστη και αναγνωρίσιμα από το Matlab.

Export
DataExport

Data
Export

Label: DataExport

Data set: Study 1/Solution 1 (sol1)

Select via: Stored output times

Time:

0
0.1
0.2
0.3
0.4
0.5

Expressions

Expression	Unit	Description
u	1	Dependent variable u
ut	1/s	Dependent variable u, first time derivative

Expression:
ut

Description:
Dependent variable u, first time derivative

Output

Filename: C:\Users\George\Γ Browse...

Always ask for filename

Points to evaluate in: Take from data set

Data format: Spreadsheet

Στη διεπιφάνεια αυτή ο χρήστης μπορεί να προσθέσει ή να αφαιρέσει μεταβλητές προς εξαγωγή καθώς και να επιλέξει χρονικές στιγμές από αυτές για τις οποίες παρήχθησαν αποτελέσματα μέσω της επιλογής range του βήματος (ζ).

ΠΑΡΑΡΤΗΜΑ Β

B1. Κώδικας για εφαρμογή διεργασίας Gauss, ενότητας (A), Κεφαλαίου II.

```

% Προηγείται εισαγωγή δεδομένων, διανυσμάτων u,ut,ux,uxx από το
χρήστη.
% Αφαίρεση τιμών NaN που αντιστοιχούν σε τίτλους και κελιά με
περιγραφή, όπως για παράδειγμα χρονικές στιγμές, από το Excel.
u=u(~isnan(u));
ut=ut(~isnan(ut));
ux=ux(~isnan(ux));
uxx=uxx(~isnan(uxx));

% Πίνακας για εκπαίδευση GP
% Επιλογή 3.000 μοναδικών σημείων από το σύνολο
k = randperm(length(u),3000);
u=u(k);
ut=ut(k);
ux=ux(k);
uxx=uxx(k);
% Σχηματισμός πίνακα
mat_1=[u ux uxx];

% Εφαρμογή Gaussian process
% Κατασκευή μοντέλου GP
mdlk = fitrgp(mat_1,ut,'KernelFunction','ardsquaredexponential');

% Έλεγχος ARD
sigmaK = mdlk.KernelInformation.KernelParameters(1:end-1,1)
el_K = log(sigmaK)

```

B2. Κώδικας για εκπαίδευση νευρωνικών δικτύων

```

% Solve an Input-Output Fitting problem with a Neural Network
% Script generated by Neural Fitting app
% Created 17-Sep-2021
%
% This script assumes these variables are defined:
%
%   mat_1 - input data.
%   mat_2 - target data.

x = mat_1';
t = mat_2';

% Choose a Training Function
trainFcn = 'trainlm'; % Levenberg-Marquardt backpropagation.

% Create a Fitting Network
H1 = 10;
H2 = 10;
% For more layers, add H3,... For less, remove H2.
net = fitnet([H1 H2],trainFcn);

% Choose Input and Output Pre/Post-Processing Functions
net.input.processFcns = {'removeconstantrows','mapminmax'};
net.output.processFcns = {'removeconstantrows','mapminmax'};

% Setup Division of Data for Training, Validation, Testing

```

```

% First Method - Mark the next 5 lines with % to suppress
net.divideFcn = 'dividerand'; % Divide data randomly
net.divideMode = 'sample'; % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
% Second Method - Unmark the next 6 lines by deleting % to use
%net.divideFcn = 'divideind'; % Divide data by index
%in_1=randperm(87480);
%net.divideMode = 'sample'; % Divide up every sample
%net.divideParam.trainInd = in_1(1:74280);
%net.divideParam.valInd = in_1(74281:87480);
%net.divideParam.testInd = 87481:160380;

% Choose a Performance Function
net.performFcn = 'mse'; % Mean Squared Error

% Choose Plot Functions
net.plotFcns = {'plotperform','plotregression'};
net.layers{1}.transferFcn = 'tansig';
net.layers{2}.transferFcn = 'tansig';

% Train the Network
[net,tr] = train(net,x,t);

% Test the Network
y = net(x);
e = gsubtract(t,y);
performance = perform(net,t,y)

% Recalculate Training, Validation and Test Performance
trainTargets = t .* tr.trainMask{1};
valTargets = t .* tr.valMask{1};
testTargets = t .* tr.testMask{1};
trainPerformance = perform(net,trainTargets,y)
valPerformance = perform(net,valTargets,y)
testPerformance = perform(net,testTargets,y)

```

B3. Κώδικες επίλυσης διαφορικής εξίσωσης με χρήση μοντέλου NN και ode solver, όπως περιγράφεται στην ενότητα (A) του Κεφαλαίου II.

```

% Ο αριθμός των σημείων στα οποία κόβω το διάστημα x ονομάζεται a.
% Προτεινόμενη τιμή 101, να αλλαχτεί και μέσα στη συνάρτηση dydt
a=101; x = linspace(0,5,a); x=round(x,2);
dx=x(2)-x(1);

% Χρόνοι στους οποίους η ode15s θα δώσει εξόδους
ts=0:0.01:2;

% Αρχικές συνθήκες y0, τιμή του y για κάθε x σε χρόνο t=to, επιλογή
% μίας από τις δύο με σβήσιμο του % πριν το y0
y0 = exp(-((x-2).^2)/(2*0.1^2));
%y0=10./(10+1*exp(10*(x-0.25)));

% Συνοριακές συνθήκες, τιμές του y για δύο τιμές του x σε κάθε t.
Στην περίπτωση Neumann/Robin είναι απλώς αρχικές συνθήκες για
x=1,x=a.
y0(1)=0;
y0(a)=0;

```

```
f=@(t,y) XronikoDiaforiko(y',dx)
[t,y] = ode15s(f,ts,yo)
```

Συνάρτηση πρόβλεψης τιμών dy/dt για συνοριακές συνθήκες Dirichlet:

```
function dydt = XronikoDiaforiko(y,dx)
% Φόρτιση μοντέλου NN
load NET7000K;
a=101;
% Memory preallocation για yxx
yxx(1)=0;
yxx(a)=0;
% Υπολογισμός yxx με πεπερασμένες διαφορές
for i=2:(a-1)
    yxx(i) = (y(i+1)+y(i-1)-2*y(i))/(dx^2);
end

% Κατασκευή πίνακα διανυσμάτων εισόδου του νευρωνικού δικτύου
trsp=[y(2:(a-1));yxx(2:(a-1))];

% Συνοριακές συνθήκες Dirichlet
dydt(1)=0;
dydt(a)=0;

% Πρόβλεψη με χρήση NN
dydt(2:(a-1)) = NET7000K(trsp)
% Το μοντέλο δέχεται στήλες, χρησιμοποιώ τον ανάστροφο πίνακα.
dydt=dydt';
end
```

Συνάρτηση πρόβλεψης τιμών dy/dt για συνοριακές συνθήκες Neumann:

```
function dydt = XronikoDiaforikoNeum(y,dx)
% Φόρτιση μοντέλου NN
load NET7000K;
a=101;
% Memory preallocation για yxx
yxx(1)=0;
yxx(a)=0;
% Υπολογισμός yxx με πεπερασμένες διαφορές
for i=2:(a-1)
    yxx(i) = (y(i+1)+y(i-1)-2*y(i))/(dx^2);
end

% Κατασκευή πίνακα διανυσμάτων εισόδου του νευρωνικού δικτύου
trsp=[y(2:(a-1));yxx(2:(a-1))];

% Συνοριακές συνθήκες Neumann,  $(dy/dx)|_{x=0} = c$ 
c = 0.1;
dydt(1)=2*(y(2)-y(1)-c*dx)/dx^2+y(1)^2;
dydt(a)=2*(y(a-1)-y(a)+c*dx)/dx^2+y(a)^2;

% Πρόβλεψη με χρήση NN
dydt(2:(a-1)) = NET7000K(trsp)
% Το μοντέλο δέχεται στήλες, χρησιμοποιώ τον ανάστροφο πίνακα.
dydt=dydt';
end
```

Συνάρτηση πρόβλεψης τιμών dy/dt για συνοριακές συνθήκες Robin:

```

function dydt = XronikoDiaforikoRob(y,dx)
% Φόρτιση μοντέλου NN
load NET7000K;
a=101;
% Memory preallocation για yxx
yxx(1)=0;
yxx(a)=0;

% Υπολογισμός yxx με πεπερασμένες διαφορές
for i=2:(a-1)
    yxx(i) = (y(i+1)+y(i-1)-2*y(i))/(dx^2);
end

% Κατασκευή πίνακα διανυσμάτων εισόδου του νευρωνικού δικτύου
trsp=[y(2:(a-1));yxx(2:(a-1))];

% Συνοριακές συνθήκες Robin
%  $(dy/dx)|_{x=0} = c1*(y|x=0 +c2)$ 
%  $(dy/dx)|_{x=5} = c3*(y|x=5 +c4)$ 
c1 = 1; c2 = 0.05;
c3 = -0.1; c4 = 2;
dydt(1)=2*(y(2)-y(1)-c1*dx*y(1)+dx*c1*c2)/dx^2+y(1)^2;
dydt(a)=2*(y(a-1)-y(a)+dx*c3*y(a)-dx*c3*c4)/dx^2+y(a)^2;

% Πρόβλεψη με χρήση NN
dydt(2:(a-1)) = NET7000K(trsp)
% Το μοντέλο δέχεται στήλες, χρησιμοποιώ τον ανάστροφο πίνακα.
dydt=dydt';
end

```

ΠΑΡΑΡΤΗΜΑ Γ.**Κώδικες επίλυσης προβλήματος μεταβατικής κατάστασης με χρήση μοντέλου NN και ode solver, όπως περιγράφεται στην ενότητα (B) του Κεφαλαίου II.**

```

% Ο αριθμός των σημείων στα οποία κόβω το διάστημα z, ονομάζεται a.
% Προτεινόμενη τιμή 51, να αλλάχτεί και μέσα στη συνάρτηση dydt
a=51; z = linspace(0,1,a);
dz=z(2)-z(1);

% Σταθερές εξισώσεων - Προσοχή: Να αλλάζονται και μέσα στη συνάρτηση
Pe = 5 ; % Αριθμός Peclet
beta = 1.5; % Αδιάστατος συντελεστής μεταφοράς θερμότητας
B = 12 ; % Αδιάστατη αύξηση αδιαβατικής θερμοκρασίας
gamma = 20; % Αδιάστατη ενέργεια ενεργοποίησης

% Χρόνος επίλυσης timespan
ts=[0 20];

% Αρχικές συνθήκες y0, τιμή του y για κάθε x σε χρόνο t=to
x1_init = 0.2;
x2_init = 2;
y0 = [ x1_init*ones(1,a) x2_init*ones(1,a) ];

f=@(t,y) XronikoDiaforikoFTTEST(y',dz)
[t,y] = ode15s(f,ts,y0)
s1=y(end,1:a);
s2=y(end,(a+1):2*a);

```

```

function dydt = XronikoDiaforikoFTTEST(y,dz)
% Φόρτιση μοντέλου NN
load NETTESTFILE160K.mat;
a=51;
Pe = 5 ; % Αριθμός Peclet
beta = 1.5; % Αδιάστατος συντελεστής μεταφοράς θερμότητας
B = 12 ; % Αδιάστατη αύξηση αδιαβατικής θερμοκρασίας
gamma = 20; % Αδιάστατη ενέργεια ενεργοποίησης
Da=0.12; % Αριθμός Damkohler
CDa=ones(1, (a-2))*Da;

% Η έξοδος της ode15s είναι στήλη με δύο διανύσματα, επιλογή κάθε ενός
x1=y(1:a);
x2=y((a+1):2*a);

% Memory preallocation για x1z,x2z,x1zz,x2zz
x1zz(a)=0; x2zz(a)=0;
x1z(a)=0; x2z(a)=0;
% Συνοριακή συνθήκη
x1z(1)=Pe*x1(1);
x2z(1)=Pe*x2(1);

% Υπολογισμός x1z,x2z,x1zz,x2zz με πεπερασμένες διαφορές
for i=2:(a-1)
    x1z(i) = (x1(i+1)-x1(i-1))/(2*dz);
    x2z(i) = (x2(i+1)-x2(i-1))/(2*dz);
    x1zz(i) = (x1(i+1)+x1(i-1)-2*x1(i))/(dz^2);
    x2zz(i) = (x2(i+1)+x2(i-1)-2*x2(i))/(dz^2);
end

% Κατασκευή πίνακα διανυσμάτων εισόδου του νευρωνικού δικτύου
y=[x1;x2];
yz=[x1z;x2z];
yzz=[x1zz;x2zz];
% Προσοχή: Το μοντέλο δέχεται [x1 x2 x1x x2x x1xx x2xx CDa] και δίνει
% [x1t x2t]
trsp=[y(:,2:(a-1)); yz(:,2:(a-1)); yzz(:,2:(a-1)); CDa ];

% Συνοριακή συνθήκη Robin
dx1dt(1)= (1/Pe)*2*(x1(2)-x1(1)-Pe*dz*x1(1))/(dz^2)-Pe*x1(1)+...
    Da*(1-x1(1))*exp(x2(1)/(1+x2(1)/gamma)) ;
dx2dt(1)= (1/Pe)*2*(x2(2)-x2(1)-Pe*dz*x2(1))/(dz^2)-Pe*x2(1)-...
    beta*x2(1)+B*Da*(1-x1(1))*exp(x2(1)/(1+x2(1)/gamma)) ;

dx1dt(a)= (1/Pe)*2*(x1(a-1)-x1(a))/(dz^2)+...
    Da*(1-x1(a))*exp(x2(a)/(1+x2(a)/gamma)) ;
dx2dt(a)= (1/Pe)*2*(x2(a-1)-x2(a))/(dz^2)- beta*x2(a)+...
    B*Da*(1-x1(a))*exp(x2(a)/(1+x2(a)/gamma)) ;

dydt(:,1)=[dx1dt(1);dx2dt(1)];
dydt(:,a)=[dx1dt(a);dx2dt(a)];

% Πρόβλεψη με χρήση NN
dydt(:,2:(a-1)) = NETTEST3020_1(trsp)

% Η ode15s δέχεται στήλες, μετατροπή πίνακα εξόδου NN που αποτελείται
% από δύο γραμμές με διανύσματα x1t και x2t σε μία στήλη.
dydt=dydt';
dydt=[dydt(:,1);dydt(:,2)];
end

```

ΠΑΡΑΡΤΗΜΑ Δ.

Κώδικες επίλυσης προβλήματος μόνιμης κατάστασης με χρήση μοντέλου NN και μέθοδο Newton-GMRES, όπως περιγράφεται στην ενότητα (Γ) του Κεφαλαίου II.

```

global x Res fn0 xpt
xpt=linspace(0,1,51);
dx=xpt(2)-xpt(1);
load VariousNETS87KSS.mat
load sol4.mat % load initial guess
x=[s1 s2 1]';
x(end)=x(end)+0.1; % x(end) is the parameter Da
% Start Newton Raphson iterations
params=[1e-2 20 3];
tol = 1e-5; % Newton iterations' tolerance

Res=FT_XDF(x(1:end-1));
np=length(x)-1;
while 1
    kmax=np;
    % Memory pre-allocation for Res
    Res=zeros(np,1);
    fn0=FT_XDF(x(1:np));
    Res=fn0;

    % call GMRES for the solution of linearized problem
    [dx,error,total_iters] = gmres ( zeros(np,1), -Res, @atv, params );

    err=max(abs(dx));
    Newton_error = num2str(err)
    % Update x
    x(1:np)=x(1:np)+dx;
    % Break Condition
    if err <= tol
        break
    end
end

end

function dydt = FT_XDF(y)
load VariousNETS87KSS;

a=51; z = linspace(0,1,a);
dz=z(2)-z(1);
% Σταθερές εξισώσεων
Pe = 5 ; % Αριθμός Peclet
beta = 1.5; % Αδιάστατος συντελεστής μεταφοράς θερμότητας
B = 12 ; % Αδιάστατη αύξηση αδιαβατικής θερμοκρασίας
gamma = 20; % Αδιάστατη ενέργεια ενεργοποίησης
Da=0.1; % Αριθμός Dahmkohler

x1=y(1:a);
x2=y((a+1):2*a);
x1zz(a)=0; % Χρησιμοποιείται ως memory preallocation για την x1zz
x2zz(a)=0; % Χρησιμοποιείται ως memory preallocation για την x2zz
x1z(1)=Pe*x1(1); % Συνοριακή συνθήκη για x1
x2z(1)=Pe*x2(1); % Συνοριακή συνθήκη για x2
x1z(a)=0; % Συνοριακή συνθήκη στο a για x1
x2z(a)=0; % Συνοριακή συνθήκη στο a για x2

```

```

for i=2:(a-1)
x1z(i) = (x1(i+1)-x1(i-1))/(2*dz);
x2z(i) = (x2(i+1)-x2(i-1))/(2*dz);
x1zz(i) = (x1(i+1)+x1(i-1)-2*x1(i))/(dz^2);
x2zz(i) = (x2(i+1)+x2(i-1)-2*x2(i))/(dz^2);
end

y=[x1 x2]';
yz=[x1z;x2z];
yzz=[x1zz;x2zz];
CDa=ones(1, (a-2))*Da;

trsp=[y(:,2:(a-1)); yz(:,2:(a-1)); yzz(:,2:(a-1)); CDa];

dx1dt(1)= (1/Pe)*2*(x1(2)-x1(1)-Pe*dz*x1(1))/(dz^2)-Pe*x1(1)+...
    Da*(1-x1(1))*exp(x2(1)/(1+x2(1)/gamma)) ;
dx2dt(1)= (1/Pe)*2*(x2(2)-x2(1)-Pe*dz*x2(1))/(dz^2)-Pe*x2(1)- ...
    beta*x2(1)+B*Da*(1-x1(1))*exp(x2(1)/(1+x2(1)/gamma)) ;

dx1dt(a)= (1/Pe)*2*(x1(a-1)-x1(a))/(dz^2)+...
    Da*(1-x1(a))*exp(x2(a)/(1+x2(a)/gamma)) ;
dx2dt(a)= (1/Pe)*2*(x2(a-1)-x2(a))/(dz^2)- beta*x2(a)+...
    B*Da*(1-x1(a))*exp(x2(a)/(1+x2(a)/gamma)) ;

dydt(:,1)=[dx1dt(1);dx2dt(1)];
dydt(:,a)=[dx1dt(a);dx2dt(a)];

dydt(:,2:(a-1)) = NET3510_2(trsp);

dydt=dydt'; % Αυτό τα κάνει από 2 γραμμές, 2 στήλες
dydt=[dydt(:,1);dydt(:,2)]; % Αυτό τα κάνει από 2 στήλες, μία στήλη
end

```